HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Computer Science and Engineering
Laboratory of Software Business and Engineering

Juha Vuojärvi

# Verkonvalvonta-ohjelmisto



Master's Thesis

Espoo, March 30, 2009

Professor Sasu Tarkoma

Hannu Tuominen, M.Sc.

| TEKNILLINEN KORKEAKOULU<br>Informaatio- ja luonnontieteiden tiedekunta<br>Tietotekniikan koulutusohjelma | DIPLOMITYÖN TIIVISTELMÄ |
|---|---|

| Tekijä | Päiväys |
|---|---|
| Juha Vuojärvi | 30.3.2009 |
| | Sivumäärä<br>59 |

| Työn nimi |
|---|
| Verkonvalvonta-ohjelmisto |

| Professuuri | Koodi |
|---|---|
| Tietoliikenneohjelmistot | T-110 |

| Työn valvoja |
|---|
| Ma professori Sasu Tarkoma |

| Työn ohjaaja |
|---|
| DI Hannu Tuominen |


Kriittisten tietoteknisten sovellusten ja -palveluiden turvaamiseksi on verkkopalveluiden toimivuus pystyttävä takaamaan. Verkonvalvontatyökalujen avulla näiden sovellusten ja palvelujen valvonta voidaan automatisoida. Koska valvonnan avulla esiintyvät häiriöt huomataan välittömästi, voidaan niiden korjaamiseen kuluvaa tarvittavaa aikaa lyhentää merkittävästi. Lisäksi verkonvalvontatyökaluilla voidaan todentaa ulkoisten palvelutoimittajien todellisia vasteaikoja ja niiden toimivuutta, sillä mahdollisista vikatilanteista saadaan yrityksen sisällä heti tieto.

Edelliset taustatekijät olivat perusteena diplomityöni aiheeksi. Diplomityöni koostuu kahdesta osasta:

- Proof-of-concept verkonvalvontatyökalun ohjelmoimisesta käyttäen avoimen lähdekoodin työkaluja ja varusohjelmia.

- Vertailla ja analysoida toteutettua verkonvalvontatyökalua kahteen jo olemassa olevaan verkonvalvontatuotteeseen: Kaupalliseen NetEye:n, joka on ohjelmisto- ja laitteistopohjainen tuote sekä ilmaiseen ohjelmistopohjaiseen Hewlett-Packard System Insight Manager-tuotteeseen

Verkonvalvonta-ohjelmisto toteutettiin Perl-ohjelmointikielellä ja sen lisämoduuleilla, Apache palvelinohjelmistolla, RRDtool ja SendMail varusohjelmistoja käyttäen.



Avainsanat

Agentti, Apache, ICMP, hallinta-ohjelmisto, NetEye, NMS, Perl, RRDtool, SIM, SNMP, System Insight Manager, Verkonvalvonta-ohjelmisto

HELSINKI UNIVERSITY OF TECHNOLOGY
Faculty of Information and Natural Sciences
Degree Programme of Computer Science and Engineering

ABSTRACT OF MASTER'S THESIS

Author
Juha Vuojärvi

Date
30.03.2009

Pages
59

Title of thesis
Verkonvalvonta-ohjelmisto

Professorship
Data Communications Software

Professorship Code
T-110

Supervisor
Professor (pro tem) Sasu Tarkoma

Instructor
Hannu Tuominen, M. Sc.

In order to provide mission critical IT-services, such as application and services one has to ensure availability and functionality relevant network services. Network Management Systems (NMS) are used for automatic monitoring of availability and functionality of these services. Via these systems service disruptions are notified immediately and time needed for troubleshooting is reduced. In addition, NMS's can be used as tolls for validating measured performance levels against service level agreement (SLA) with internal and external parties.

Diploma work consists two parts:

- Proof-of-concept NMS programming using open source software
- Comparison and analysis of programmed NMS tools against two already existing NMS tools. Comparison and analysis was done against commercial NetEye-software and hardware based solution and software based Hewlett-Packard System Insight Manager

Own NMS was implemented using Perl-programming language and its modules and using Apache-server, RRDTool- and SendMail software.

Keywords

Agent, Apache, ICMP, NetEye, network management systems, NMS, Perl, RRDtool, SIM, SNMP, System Insight Manager

# Sisällysluettelo

# Kuvaluettelo

# Esimerkkiluettelo

# 1. Johdanto

Kriittisten tietoteknisten sovellusten ja -palveluiden toimivuuden turvaamiseksi on verkkopalveluiden toimivuus pystyttävä takaamaan. Verkonvalvontatyökalujen avulla näiden sovellusten ja palvelujen valvonta pystytään automatisoimaan. Koska häiriöt huomataan välittömästi, voidaan palveluissa mahdollisesti esiintyvien häiriöiden korjaamiseen tarvittavaa aikaa lyhentää. Verkonvalvontatyökaluilla voidaan todentaa ulkoisten palvelutoimittajien todellisia vasteaikoja ja niiden toimivuutta, sillä mahdollisista vikatilanteista saadaan yrityksen sisällä heti tieto.

Edelliset taustatekijät olivat perusteena diplomityöni aiheeksi. Diplomityöni koostuu kahdesta osasta:

- Proof-of-concept verkonvalvontatyökalun ohjelmoimisesta käyttäen avoimen lähdekoodin työkaluja ja varusohjelmia.

- Vertailla ja analysoida toteutettua verkonvalvontatyökalua vähintään yhteen jo olemassa olevaan kaupalliseen tai avoimeen lähdekoodiin perustuvaan tuotteeseen

Diplomityön toteutus, testaus ja siihen liittyvien tuotteiden vertailut suoritettiin erillisessä eristetyssä testiympäristössä, jotta Altian tietoverkkoon ei kohdistuisi mitään häiriötekijöitä tai ulkoisia riskitekijöitä. Verkonvalvonta-ohjelmisto toteutettiin Perl-ohjelmointikielellä ja sen lisämoduuleilla, Apache palvelinohjelmistolla, RRDtool ja SendMail varusohjelmistoja käyttäen. Tietoturvasyistä johtuen kaikki Altian laitteisiin liittyvät identifioivat tiedot on anonymisoitu tai muutettu.

Kappaleessa kaksi esitellään mihin ja miksi verkonvalvonta-ohjelmistoja käytetään sekä kuvataan niiden liittyviä ongelmia sekä rajoitteita. Lisäksi esitellään lyhyesti muutamia verkonvalvonta tuotteita. Kappaleessa kolme kuvataan verkonvalvonta-ohjelmistojen tekninen toteutus yleistasolla sekä kuvataan toteutuksissa yleisimmin käytetyt tietoliikenneprotokollat sekä esimerkkitapauksin mihin niitä käytetään verkonvalvonta-ohjelmistoissa. Kappaleessa neljä kuvataan verkonvalvonta ohjelmiston ohjelmistokehityksen eri vaiheita. Kappaleessa viisi käydään läpi diplomityössä ohjelmoitu verkonvalvontatyökalun tekninen toteutus sekä työkalut ja ohjelmistot, joita sen toteuttamiseen käytettiin. Lisäksi kappaleessa kuvataan yksityiskohtaisemmin jokainen ohjelmamoduuli, sekä kuvataan esimerkeillä niiden toiminnallisuutta. Kappaleessa kuusi analysoidaan toteutuksen onnistumista sekä kerrotaan esimerkkejä käyttäen miten verkonvalvonta-ohjelmiston testaus suoritettiin eri ohjelmistokehityksen vaiheissa. Kappaleessa seitsemän analysoidaan miten diplomityössä ohjelmoidun verkonvalvontatyökalun toteutus onnistui vaatimusmäärittelyä vasten sekä verrataan ohjelmiston toteutusta ja toiminnallisuutta valittua verrokkituotetta vastaan. Lisäksi kappaleessa käydään läpi tämän hetken trendit verkonvalvontatuotteissa sekä listataan kehityskohteet oman verkonvalvonta-ohjelmiston osalta. Kappaleessa kahdeksan suoritetaan yhteenveto saatujen tulosten osalta ja diplomityön osalta.

# 2. Verkonvalvonta-ohjelmistot

Verkonvalvonta-ohjelmistojen (NMS, Network Management Systems) avulla pyritään automatisoimaan ja yksinkertaistamaan tietoverkkojen valvontaa. Toteutuksesta riippuen verkonvalvonta-ohjelmiston avulla voidaan valvoa esimerkiksi tietoverkkojen, verkkopalvelujen ja -sovellusten toimintaa, verkkolaitteiden kuormitusta, käyttäjälle tarjottavien verkkopalveluiden tai sovellusten vasteaikoja [8-15, 40]

## 2.1 Mihin verkonvalvonta-ohjelmistoja käytetään

Verkonvalvontaohjelmistoja käytetään yleisesti sekä yritysten oman IT-ympäristön valvontaan että ulkoistuskumppanien ja palveluntarjoajien palveluiden laadunvalvontaan ja vianselvitykseen asiakkaan toimesta. Viimeksi mainitussa skenaariossa verkonvalvonta-ohjelmistolla voidaan verrata, analysoida ja raportoida havaitut poikkeamat sovituissa palvelutasoissa ulkoistuskumppaneille ja palveluntarjoajille.

Virhetilanteiden raportointia tai niistä tiedottamista voidaan automatisoida valvonta-ohjelmistojen avulla [8]. Virhetilanteista voidaan lähettää sähköposti- tai tekstiviesti ennalta määritellyille henkilöille.

Valvonta on proaktiivista tai reaktiivista. Proaktiivista valvontaa on esimerkiksi se, että jatkuvasti virheistä raportoiva laite tai sen osa korvataan, ennen kuin virheistä aiheutuu ongelmia. Reaktiivista verkonvalvontaan on esimerkiksi se, että verkonvalvonta-ohjelmiston avulla havaituista vioista tai ongelmista lähetetään sähköpostihälytys, ja vikatiedot tallennetaan järjestelmään tarkempaa lisäselvitystä ja -raportointia varten.

## 2.2 Miksi verkonvalvonta-ohjelmistoja käytetään

Manuaalisesti suoritettu verkonvalvonta vaatii erittäin paljon aikaa sekä osaamista [31]. Verkonvalvonta-ohjelmistojen avulla on mahdollista vähentää valvontaan tarvittavaa aikaa, ja siirtää olemassa olevien resurssien käyttö muihin tarvittaviin tehtäviin. Verkonvalvonta-ohjelmistot käyttävät standardiprotokollia, joten valvonta- ja raportointitietoja on mahdollista kerätä eri valmistajien laitteista ja sovelluksista. Kerättäviä valvontatietoja ovat muun muassa prosessorin kuormitus, vapaana olevan keskusmuistin määrä [40] sekä verkkoliikenteeseen liittyvät parametrit verkkolaitteilta, joiden avulla voidaan huomata poikkeamat verkkoliikenteessä [34, 42].

Mikäli valvonta-ohjelmassa toteutus sisältää graafisen käyttöliittymän, voidaan sen kautta tarkistaa nopeasti valvottavan tietoverkon laitteiden ja palveluiden toimivuus. Valvonta-ohjelmistot on pyritty toteuttamaan niin, että valvontaa suorittavan henkilön ei tarvitse omata suurta tietämystä esimerkiksi valvottavista laitteista tai ohjelmistoista havaitakseen ja raportoidakseen niissä esiintyvät virheet.

Useat valvonta-ohjelmiston tehtävistä voidaan automatisoida ja ajastaa, jolloin valvontatyökalun käyttöön ei normaalitilanteessa tarvitse puuttua. Esimerkiksi havaituista kriittisistä virheistä tai ongelmista informoidaan yleensä automaattisesti teksti- ja sähköpostiviestillä ennalta määritellyille henkilöille.

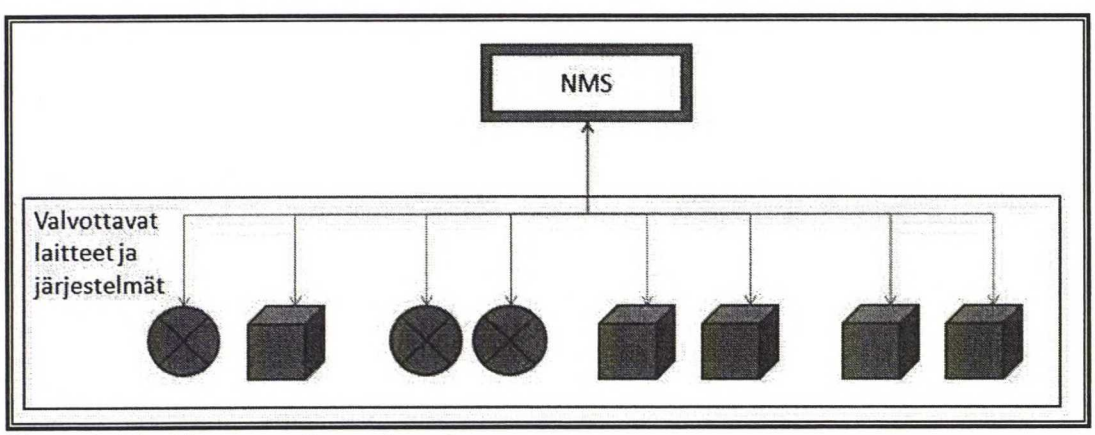
Raportointi kuuluu olennaisesti valvonta-ohjelmistojen toiminnallisuuteen ja raporttien osalta käyttäjillä on mahdollisuus koota ohjelman keräämistä tiedoista haluamansa yksityiskohdat tai tiedot. Kaupallisten verkonvalvontatuotteiden osalta on mainittavana myös niiden integroituminen muiden valmistajien valvontatyökaluihin.

## 2.3 Verkonvalvonta-ohjelmistoihin liittyvät ongelmat

### 2.3.1 Toiminnalliset ongelmat

Verkonvalvonta-ohjelmiston toiminta voi aiheuttaa valvottavassa tietoverkossa ongelmia. Ongelmat voivat johtua valvonta-ohjelmiston väärin tai virheellisesti määritellyistä asetuksista tai verkonvalvonnan väärästä toteutuksesta.

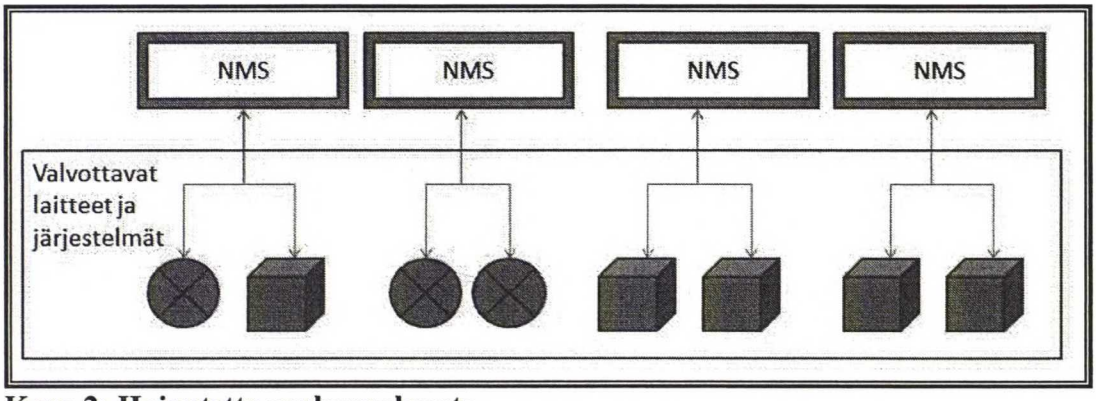
Liian usein lähetettävä SNMP (Simple Network Management Protocol) kysely kaikille valvottavan verkon laitteille aiheuttaa paljon verkkoliikennettä valvonta-ohjelmiston ja valvottavien laitteiden ja palveluiden välillä. Edellisen korjaamiseksi SNMP-protokollan versioon kaksi on lisätty tuki kyselylle, jossa yhden kyselyn vastauksena saadaan vastausjoukkoja valvottavalta laitteelta ja täten minimoitua verkkoon lähetettävien kyselyiden määrä. Korjauksella on suuri vaikutus sillä suurin osa kaupallisista verkonvalvontatuotteista käyttää SNMP kyselyitä laitteiden tilatietojen hakemiseen [29, 30].




**Kuva 1: Keskitetty verkonvalvonta**

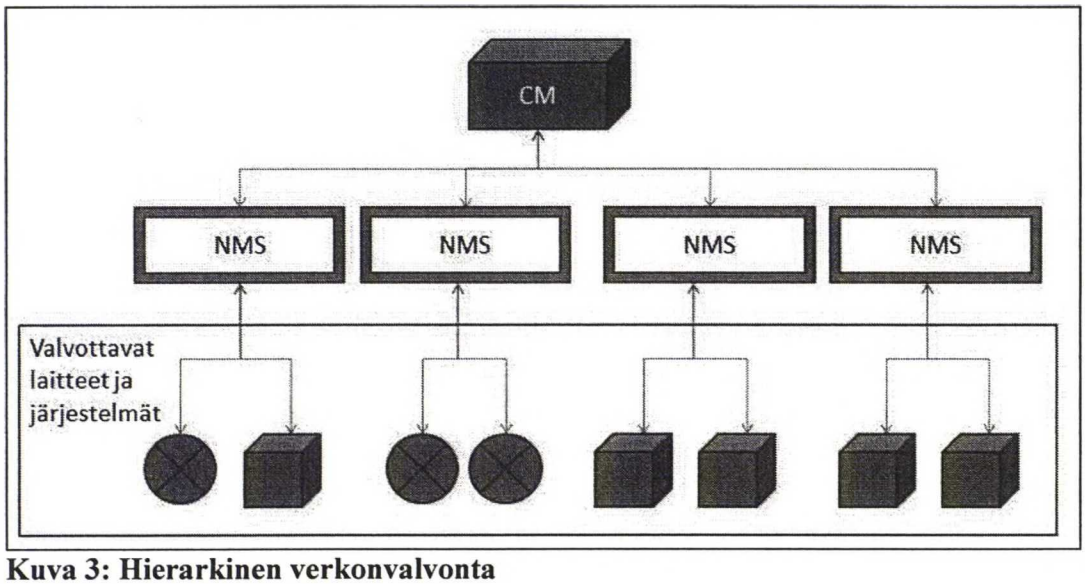
Keskitettyjen verkonvalvonta-ohjelmistojen (Kuva 1) käytössä saattaa esiintyä suorituskykyyn liittyviä ongelmia, sillä siinä yksi keskitetty valvontaohjelmisto valvoo kaikkia määriteltyjä verkon laitteita ja järjestelmiä. Mikäli valvottavia kohteita on paljon

voi valvonta-ohjelmiston toiminta ylikuormittaa verkkolaitteita tai -yhteyttä hakemalla niiden tilatietoja liian usein tai liian kattavasti [30, 40]. Ongelma voidaan välttää usealla eri tavalla: Vaihtamalla pullonkaulan muodostavat laitteistot tai tietoverkkoyhteydet, rajoittamalla valvottavien laitteistoparametrien määrää tai muuttamalla valvottavien laitteiden kyselytiheyttä harvemmaksi. Toinen mahdollinen ratkaisu on toteuttaa valvonta käyttäen hajautettua tai hierarkista verkonvalvontaa. Tällöin valvonnassa käytetyt kyselyt lähetetään useista eri laitteista (Kuvat 2-3) [30]. Hajautetun verkonvalvonnan etuna on yksittäisten verkkolaitteiden tai -linkkien ylikuormituksen minimointi. Haittoja ovat ratkaisusta aiheutuvat lisäkulut sekä keskitetyn valvonnan menetys.




**Kuva 2: Hajautettu verkonvalvonta**

Hierarkinen verkonvalvonta (Kuva 3), joka on yhdistelmä keskitetystä ja hajautetusta verkonvalvonnasta, toteutetaan kahden tason Verkonvalvonta-ohjelmistoilla. Alemman tason verkonvalvonta-ohjelmistot lähettävät kyselyitä valvottaville laitteille sekä vastaavat ja välittävät tiedot ylätason verkonvalvonta-ohjelmalle (CM, Central Manager) [30]. Haittoja ovat toteutuksen monimutkaisuus sekä siitä aiheutuvat suuret kulut.




**Kuva 3: Hierarkinen verkonvalvonta**

Verkonvalvonnan piirissä olevien laitteiden ohjelmistoversiota tai -asetuksia voidaan muuttaa verkonvalvonta-ohjelmistojen, kuten HP (Hewlett Packard) SIM (System Insight Manager), kautta [8-13]. Mikäli toiminteessa on ohjelmointivirhe, voi valvonta-ohjelmiston kautta suoritettu päivitys aiheuttaa päivitettävän laitteen toimintahäiriön, joka voidaan korjata vain paikanpäällä.

### 2.3.2 Tietoturva ongelmat

Verkonvalvonta-ohjelmistojen tietoturvaan liittyvät asiakohdat on syytä huomioida toteutuksessa, jotta pääsynvalvonnasta ja tietojen luottamuksellisuudesta huolehditaan asianmukaisesti. Edellä kuvatut vaatimukset ovat yleisesti toteutettu:

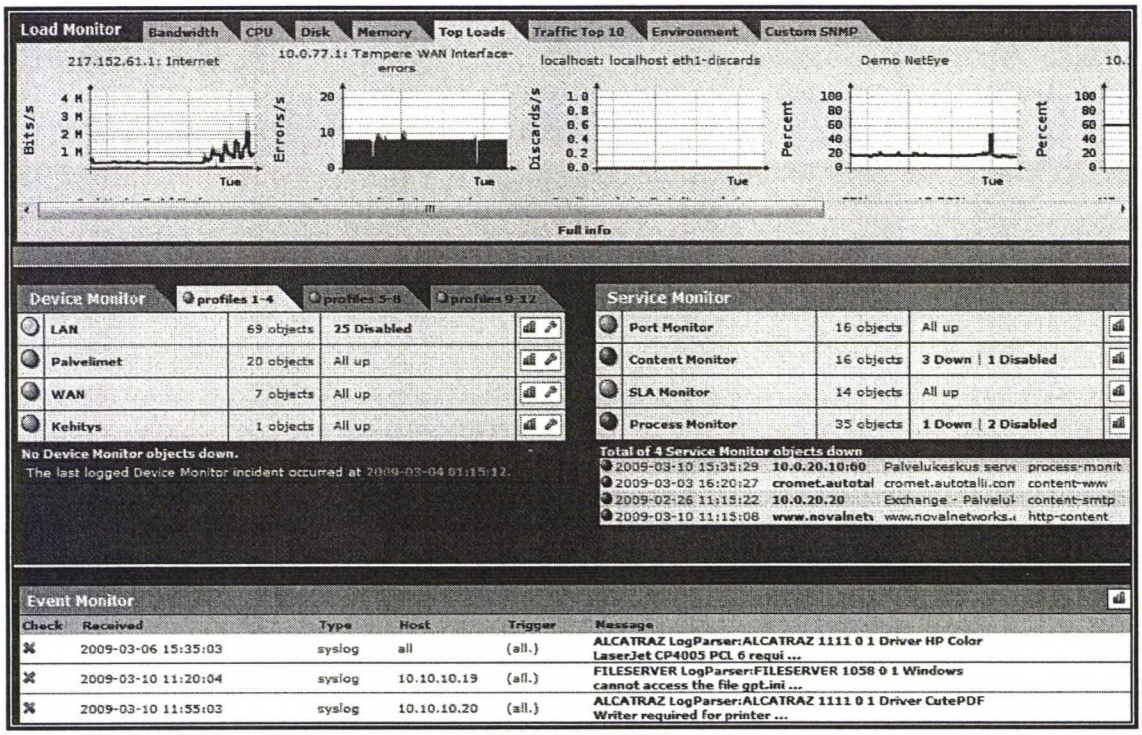
- Suojaamalla verkonvalvonta-ohjelmiston hallintayhteys
- Toteuttamalla käyttäjätunnistus verkonvalvonta-ohjelmistoon
- Ottamalla käyttöön SNMPv3, jossa tuki SNMP liikenteen salaukseen [38]

## 2.4 Verkonvalvonta-ohjelmistojen rajoitteet

Verkonvalvonta-ohjelmistojen, kuten ei muidenkaan valvontaohjelmistojen avulla, voida ratkaista kaikkia tietoverkon tai sen kautta käytettävien palveluiden ongelmia. Liiallinen luottamus tai riippuvuus verkonvalvonta-ohjelmiston toimivuuteen voi tuottaa ongelmia. Mikäli tietoverkossa tai sen palveluissa esiintyy ongelmia ja verkonvalvonta-ohjelmisto ei toimi oikein, ei vikaa tunnisteta tai siihen ei reagoida oikein [31]. Edellä kuvatusta tilanne toteutuu, kun verkonvalvonta-ohjelmisto havaitsee valvottavassa sähköpostipalvelussa ongelman ja yrittää automaattisesti lähettää sähköpostilla vikaviestiä. Kyseessä on toiminnallinen virhe, sillä vikailmoitus ei välity oikein koska verkonvalvonta-ohjelmisto yrittää käyttää vikaantunutta palvelua.

## 2.5 Verkonvalvonnan tuotteita

Verkonvalvontatuotteita on toteutettu sekä ohjelmistoratkaisuina että ohjelmisto- ja laitteistopohjaisena ratkaisuna. Esimerkkinä ohjelmistoratkaisuna toteutetusta verkonvalvontatuotteesta on HP:n SIM [8-13], jota käytin diplomityössäni vertailutuotteena omaa verkonvalvontaohjelmistoa vastaan. Esimerkkeinä kaupallisesta ohjelmisto- ja laitteistopohjaisena ratkaisusta on NetEye jota käytin diplomityössäni toisena vertailutuotteena omaa verkonvalvontaohjelmistoa vastaan (Kuva 4) [14–15, 45]. Esimerkkinä avoimeen lähdekoodiin perustuvasta tuotteesta on ohjelmistopohjainen OpenNMS verkonvalvonta-ohjelmisto [16, 17].




**Kuva 4. NetEye verkonvalvonta-ohjelmiston pääsivu**

# 3 Verkonvalvonta-ohjelmiston tekninen toteutus

Verkonvalvonta-ohjelmistot koostuvat yleensä verkonvalvonta-ohjelmistosta, agentti-ohjelmistosta sekä hallinta-ohjelmistosta riippumatta toteutuksessa käytetystä arkkitehtuuriratkaisusta [1, 8-13, 35]. Verkonvalvonta-ohjelmistoja tai -järjestelmiä kategorisoidaan monitorointitekniikan toteutuksen mukaisesti pakettikaappaukseen, SNMP-protokollaan tai tietovirran laskentaan perustuviin [34].

## 3.1 Verkonvalvonta-ohjelmisto

Valvonta-ohjelmiston tehtävänä on valvoa määriteltyjen tietoverkon laitteiden ja palveluiden toimivuutta, sekä tallentaa kerätyt tilatietoja jatkoanalysointia varten.

Käytettävyyden varmentamiseksi ohjelmisto asennetaan käyttöä varten varatuille laitteistolle. Mikäli verkonvalvonta-ohjelmisto käyttää tietokanta-ohjelmistoa, voi se toteutuksesta riippuen sijaita joko samassa laitteistossa valvonta-ohjelmiston kanssa tai erillisellä tietokantapalvelimella. Tietokantaa käytetään valvottavien laitteiden tilatietojen tallentamista, raportointia ja analysointia varten Olennainen osa verkonvalvonta-ohjelmiston toteutusta on pääsynvalvonta. Pääsynvalvonnan avulla ohjelmiston käyttö rajoitetaan vain tunnistettuihin ja autorisoituihin käyttäjiin. Lisäksi erilaisilla käyttäjätasoilla tai -ryhmillä voidaan rajoittaa halutusti pääsyä ohjelmiston eri toimintoihin.

## 3.2 Agentti-ohjelmisto

Agentti-ohjelmisto asennetaan kaikkiin laitteisiin, joita valvotaan SNMP-protokollan avulla. Agentti-ohjelmisto kommunikoi verkonvalvonta-ohjelmiston kanssa ohjelmoidun mukaisesti. Toteutuksesta riippuen valvonta on proaktiivista tai reaktiivista. Proaktiivisessa valvonnassa valvottava laite voi raportoida määriteltyjen laitteistoparametrien ylittymisestä valvonta-ohjelmistolle SNMP Trap-sanoman avulla. Reaktiivisessa valvonnassa valvonta-ohjelmisto hakee valvottavilta laitteilta tilatiedot määriteltyjen asetusten mukaisesti.

## 3.3 Hallinta-ohjelmisto

Verkonvalvonta-ohjelmiston hallinta-ohjelmisto voidaan yleensä asentaa sellaisiin laitteisiin, jotka tukevat Java-ohjelmistoa tai joissa on WWW-selain. Edellisen kaltaisia laitteita ovat työasemat, palvelimet ja erilaiset mobiili-laitteet. Valvottavat laitteet, sovellukset ja palvelut sekä erilaiset toiminteet havaituissa vika- tai virhetilanteissa määritellään yleensä hallinta-ohjelmiston kautta. Toteutuksesta ja käyttöoikeuksista riippuen hallinta-ohjelmiston kautta voidaan myös seurata raportointia valvottavien laitteiden, käytettävyyden ja havaittujen virheiden ja vikojen osalta.

## 3.4 Miten verkonvalvonta toteutetaan

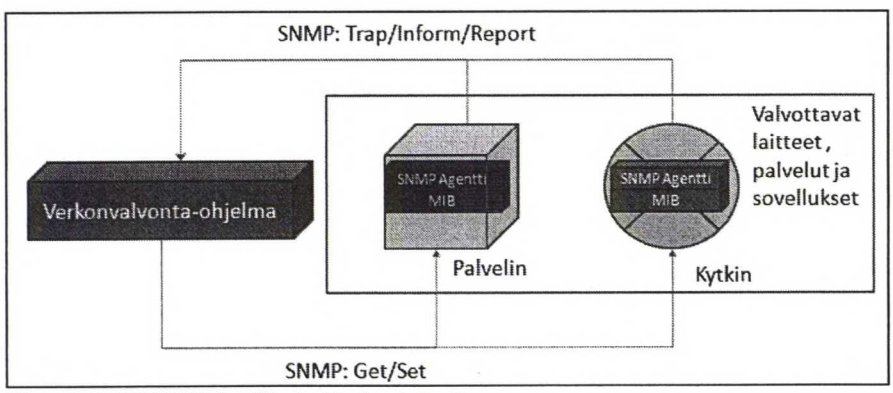
Verkonvalvonta-ohjelmistojen toiminnallisuuden toteutukseen käytetään yleisesti seuraavia standardiprotokollia sekä niihin sisältyviä työkaluja ja komentoja [41].

### 3.4.1 SNMP

Suurin osa verkonvalvonta-ohjelmista käyttää SNMP-tietoliikenneprotokollaa laitteiden ja palveluiden valvontaan [30, 32]. Protokollan avulla on mahdollista sekä valvoa että hallita lähes kaikkia verkkoon kytkettyjä laitteita. SNMP protokollasta on julkaistu kolme versiota, jotka kuvataan lyhyesti alla olevissa kappaleissa.

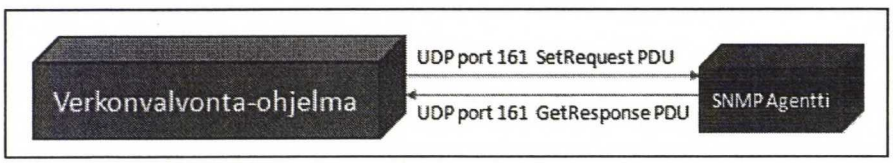
SNMP-hallintajärjestelmä koostuu seuraavista (Kuva 5) [35]:

- SNMP-agentista (Agent): Yhdestä tai useammasta laitteesta, joihin on asennettu agentti-ohjelmisto
- SNMP-valvonnasta (Manager): Sisältää ohjelman, jonka kautta viestejä voidaan lähettää ja vastaanottaa hallittavilta agenteilta (Kuvan 5 verkonvalvonta-ohjelmisto sisältää tämän toiminnallisuuden)
- Hallintaprotokollasta (SNMP), jonka avulla hallintatietoja voidaan siirtää valvonta-ohjelmiston ja valvottavien laitteiden välillä.




**Kuva 5: SNMP-hallintajärjestelmä**

SNMP PDU (Protocol Data Unit) sanomilla verkonvalvonta-ohjelma voi hakea verkkolaitteilta niiden tilatietoja ja asetuksia tai muuttaa niitä. Protokollan tiedonsiirtoon käytetään yhteydetöntä UDP-protokollaa (User Datagram Protocol). UDP porttia 161 käytetään sekä kysely- (Get) että asetusoperaatioihin (Set) (Kuva 6).
Toiminnallisesti asetukset suoritetaan muuttamalla laitteen tietyn asetuksen arvoa, esimerkiksi vaihtamalla reitittimen tietyn verkkokortin nopeusparametrin arvo 100000:sta 10000:n, eikä lähettämällä vastaavaa komentoa. SNMP asetusoperaatiot ovat atomisia, ne joko tehdään onnistuneesti tai niitä ei tehdä ollenkaan.

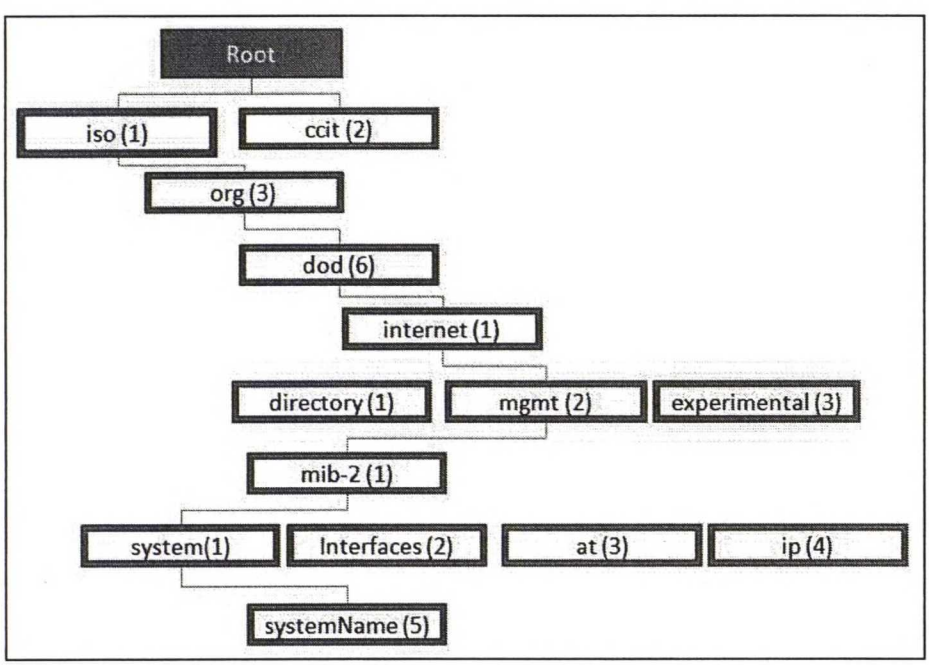



**Kuva 6: SNMP viestinvälitys hallinta-ohjelmiston ja agentin välillä**

Verkkolaitteet voivat lähettää tietoja valvonta-ohjelmistolle PDU hälytys (Trap)- tai tiedotesanomilla (Inform/Report) käyttäen tiedonsiirtoon UDP-porttia 162.

SMI (Structure of Management Information) määrittelee puumaisesti tietorakenteen jota SNMP käyttää [20]:

- Nimi, Object Identifier (OID): Esitetään numeroilla tai tekstimuodossa. Esimerkkinä systemName-muuttuja voidaan esittää 1.3.6.1.2.1.1.5.0 tai iso.org.dod.internet.directory.mib-2.system.systemName hakuna (Kuva 7).
- Tyyppi ja syntaksi: SMI käyttää laitteistoriippumatonta Abstract Syntax Notation One (ASN.1), joka määrittelee nimet ja muuttujien sallitut tyypit MIB:ssä
- Koodaus: Jakaa objektin Basic Encoding Rules (BER) mukaisesti oktetteihin.




**Kuva 7: Object Identifier-tietorakenne**

MIB (Management Information Base) on hallintatietokanta jossa määritellään: laitteelle talletettavat tiedot, sallitut komennot ja operaatiot, sekä määrittelee edellisten tarkoituksen ja sisällön. Laitevalmistajat julkaisevat omia MIB hallintatietokantojaan, jotka sisältävät kaikki laitteiden tukemat ominaisuudet joita ei ole määritelty standardi MIB:n. Loogisesti MIB-hallintatietokanta sijaitsee valvottavalla laitteella, agentissa, ja sinne tallennetaan laitteen tiedot. Valvonta-ohjelmisto lähettää SNMP protokollaa

käyttäen kyselyn SNMP-agentille joka hakee pyydetyt tiedot MIB-hallintatietokannasta. Agentti palauttaa vastauksen valvontaohjelmistolle SNMP-protokollaa käyttäen.

Mikäli valvottaville laitteille asennetaan laitevalmistajien MIB-tietokantoja, on vastaavat MIB-tietokannat asennettava myös valvonta-ohjelmistoon. Ilman edellä kuvattuja asennuksia laitevalmistajan tekemät laajennukset eivät näy SNMP-hauissa oikein.

Vuonna 1988 julkaistu ensimmäinen SNMP versio (SNMPv1) toteutus sisältää viisi operaatiota

- GET: Haetaan yksi nimetty tietokenttä
- GETNEXT: Haetaan seuraava tietokenttä
- SET: Muutetaan nimetyn tietokentän arvo
- TRAP: Laite raportoi muuttuneesta tilatiedosta

SNMP luottosuhteen määrittämiseen valvonta-ohjelmiston ja valvottavan laitteen agentin välillä käytetään yhteisönimeä (Community name). Valvottavassa laitteessa luottosuhde voidaan määrittää kolmeen käyttöoikeustasoon: kirjoitus (read/write), luku (read) tai viestinvälitys (trap/inform/notification/report). Mikäli käytössä on vain lukuoikeus, valvonta-ohjelmisto voin vain lukea valvottavan laitteen tietoja. SNMPv1 toteutus tukee vain 32-bittisiä tietokenttiä ja tiedonsiirto valvonta-ohjelmiston ja valvottavien laitteiden välillä siirretään selkokielisenä [20].

SNMP versio kahden (SNMPv2) toteutus sisältää kaikki SNMPv1 operaatiot sekä seuraavat lisäoperaatiot:

- GETBULK REQUEST - Yhden kyselyn vastauksena saadaan vastausjoukkoja valvottavalta laitteelta.
- INFORM - Kuten Trap mutta vastaanottajan täytyy vastata viestiin Response-kuittausviestillä
- NOTIFICATION: Valvottava laite ilmoittaa valvonta-ohjelmistolle muuttuneesta tilasta.

SNMPv2 toteutus tukee sekä 32- ja 64-bittisiä tietokenttiä [1, 20].

SNMP versio kolmen (SNMPv3) toteutus sisältää kaikki SNMPv1 ja SNMPv2 operaatiot sekä lisäyksiä, joista suurimmat muutokset ovat tietoturvaan liittyviä. Luottosuhteiden sijaan SNMPv3 tukee vahvempaa tunnistusta sekä mahdollistaa salauksen käytön tiedonsiirrossa seuraavasti [38]:

- Ei tunnistusta eikä salausta
- Tunnistus, mutta ei salausta
- Tunnistus ja salaus

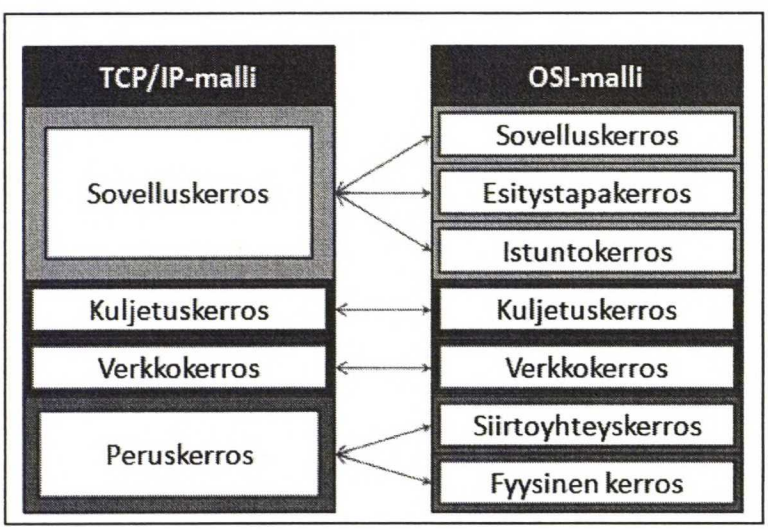
SNMPv3 salauksen osalta tuki vahvemmille Advanced Encryption Standard (AES) ja Triple Data Encryption Standard (3-DES) salausalgoritmille vuonna 2004 [37]

alkuperäisen Data Encryption Standard (DES) lisäksi [36]. Tunnistuksen osalta tuetaan sekä Message Digest (MD5) että Secure Hash Algorithm (SHA1) algoritmeja [36].

Muita merkittäviä muutoksia SNMPv3 osalta on REPORT-lisäoperaatio, jonka avulla valvonta-ohjelmistot voivat viestittää toisilleen. SNMPv3 toteutus tukee sekä 32- ja 64-bittisiä tietokenttiä.

### 3.4.2 RMON

Ehkä tärkein SNMP-protokollaan tehdyistä lisäosista on RMON (Remote Network Monitoring), sillä se tarjoaa laajennuksen protokollan toiminnallisuuteen. RMON:sta on kaksi versiota, joista läpi käydään vain uudempi versio kaksi.




**Kuva 8: TCP/IP ja OSI mallien vertailu**

RMON versio kahden avulla tietoliikennettä voidaan valvoa ja analysoida OSI-mallin (Open Systems Interconnection) tasolta kolme ylöspäin (Kuva 8) [44]. TCP/IP mallin mukaisesti siirrettävä tieto on verkko-, kuljetus ja sovelluskerroksen siirtämää tietoa. Näin ollen verkkoliikennettä voidaan esimerkiksi kategorisoida ja analysoida sisäverkko tai ulkoverkkoliikenteeksi tai tarvittaessa sovellus- tai konekohtaisesti. Koska verkonvalvonta on mahdollista hyvinkin tarkalla tasolla, voidaan RMON version kaksi laajennettua toiminallisuutta hyödyntää etenkin verkonvalvonnan proaktiivisessa seurannassa sekä tarvittavien muutostoimien analysoinnin apuvälineenä.

### 3.4.3 ICMP

Yleisimmin käytetyistä verkkoon kytkettyjen laitteiden toiminnan tarkistamiseen käytettävistä työkaluista ovat ICMP-protokollaan (Internet Control Message Protocol) sisältyvän Ping- ja Tracert-ohjelmien käyttö [1,43]. Yksinkertaisella, tehokkaalla ja

nopealla Ping-komennolla voidaan helposti todeta verkkoon kytkettyjen laitteiden vasteaika kuormittamatta verkkoa, sillä yhdessä datapaketissa siirtyy tietoa vain 32-bittiä. Lisäetuna on, että kyseinen Ping-komento sisältyy standardin mukaan TCP/IP protokollan (Transmission Control Protocol/Internet Protocol) mukaisesti kaikkiin sen toteutuksiin. Mikäli vastaanottavalta laitteelta ei saada vastausta Ping-komennon timestamp-kenttään, saadun vastauksen perusteella ei tiedetä, onko vika esimerkiksi reitityksessä tai laitevika lähteen ja kohteen välillä.

Yksinkertaistettu kuvaus Ping-komennon eli ICMP Echo_request ja echo_reply toiminnasta (Esimerkki 1)

- A Lähettää ICMP ECHO_REQUEST paketti kohdelaitteelle B Ping-komennolla "Ping B", jossa B on vastaanottajan IP-osoite. Komennon lähetysaika tallennetaan muistiin.
- Mikäli kohdelaite B löytyy verkosta ja se voi vastata, palauttaa se ECHO_REPLY paketin takaisin lähettäjälle A
- Lähettäjä A saa ECHO_REPLY-paketin. ECHO_REQUEST ja ECHO_REPLY viestin perusteella lasketaan kokonaisaika, joka kului paketin lähettämisestä vastauksen saamiseen (timestamp tiedosta).
- Tarvittaessa vaiheiden 1-2 ja 2-3 välissä suoritetaan pakettien reititystoimenpiteet.

**Esimerkki 1: Yksinkertaistettu kuvaus Ping-komennon toiminnasta**

Ping-komennon lisäparametreja käyttämällä voidaan selvittää muun muassa prosentuaalinen pakettihävikki (packet loss) yhteyttä testatessa. Tällöin lähetetään esimerkiksi 20 Echo Request-pakettia kohteeseen ja vastauksen loss-parametrin arvosta käy ilmi kuinka monta prosenttia paketeista on hävinnyt matkalla.

Tracert- tai Traceroute-komennolla voidaan tarkistaa mitä reittiä paketit välittyvät lähettäjältä vastaanottajalle. Vastauksesta nähdään myös pakettien välitykseen kuluva aika jokaisen välitykseen osallistuvan laitteen osalta.

### 3.4.4 TCP

IP-protokollan ollessa yhteydetön ja epäluotettava, tarjoaa TCP-protokolla luotettavan ja yhteydellisen tavan siirtää tietoa. TCP-protokollan toteutus sisältää valvonta- ja signalointiviestejä, joiden avulla tietoliikenteen varmuutta on parannettu esimerkiksi viestin uudelleenlähetyksen avulla [1]. Verkonvalvonnassa TCP-protokollaa käytetään yleisesti palvelujen ja laitteiden yleisen toiminnan tarkistamiseen sekä esimerkiksi järjestelmän päälläoloaikatiedon hakemiseen.

### 3.4.5 DNS

DNS (Domain Name Service) palvelun kautta laitteiden IP-osoitteet voidaan muuttaa tekstimuotoon tai päinvastoin käyttäen apuna hajautettua tietokantaa. Hajautettuna tietokantana toimivat nimipalvelimet, joihin otetaan yhteys tietoliikenneverkon kautta. DNS protokollan avulla voidaan tarkistaa myös laitteille varatut nimi- ja IP-osoiteparit sekä valvoa nimipalveluiden ja -palvelinten toimivuutta [1].

### 3.4.6 NetBIOS

Verkonvalvonta-ohjelmistot voivat käyttää myös NetBIOS (Network Basic Input Output System) protokollaa erilaisten palveluiden tai sovellusten tilatietojen tarkistamiseen sekä yksittäisten parametrien lukemiseen. Valvottava järjestelmä asettaa rajoitteita saatavien NetBIOS kyselyille, esimerkiksi palveluiden tai sovellusten tilatietojen tarkistus voidaan suorittaa onnistuneesti vain Microsoft-käyttöjärjestelmistä. Kaikista järjestelmistä voidaan tarkistaa NetBIOS kyselyillä jaettujen levyalueiden tiedot sekä järjestelmän käyttäjät.

### 3.4.7 ARP

ARP (Address Resolution Protocol) protokollan avulla voidaan liittää yhteen tietyn valvottavan verkkolaitteen käyttämä IP-osoite ja tiedonsiirtoon käytetyn verkkokortin MAC (Media Access Control) osoite. ARP-protokollan avulla voidaan näin ollen tunnistaa 24-porttisen reitittimen vikaantuessa, voidaan missä portissa vika esiintyi tai ilmenee [1].

### 3.4.8 Socket

Verkonvalvonnassa socket-tiedonvälitysabstraktiota voidaan käyttää erilaisten sovellusten toiminnallisuuden valvontaan. Socket-abstraktion etuna on, että niiden kautta voidaan siirtää mitä tahansa tietoa [1]. Socket tiedonvälitykseen käytetään yhteydellistä TCP-protokollaa tai yhteydetöntä UDP-protokollaa. Mikäli sovelluksia halutaan valvoa socket:n avulla, joudutaan valvontaan sisällyttämään lisätoiminnallisuutta, koska virhetilanteissa socketien kautta ei välity tietoa onko vika valvottavassa palvelussa, palvelimessa vai tiedonvälitykseen käytettävissä laitteissa tai verkossa.

Esimerkiksi Apache HTTP-palvelinohjelman (Hypertext Transfer Protocol) toiminnan tarkistaminen voidaan suorittaa yksinkertaisen socket kutsun avulla, sillä Apache sovellus on luonut ja sitonut socketin HTTP-standardin mukaiseen porttiin 80. Lisäksi se kuuntelee ja vastaa, mikäli tähän porttiin tulee liikennettä. Verkonvalvonta-ohjelmisto avaa uuden socketin, yhdistää sen Apache palvelimen sockettiin porttiin 80 ja tekee

tarvittavan kyselyn. Mikäli Apache-palvelinohjelmisto toimii, vastaa se verkonvalvonta-ohjelmiston lähettämään kyselyyn.

### 3.4.9 SMTP

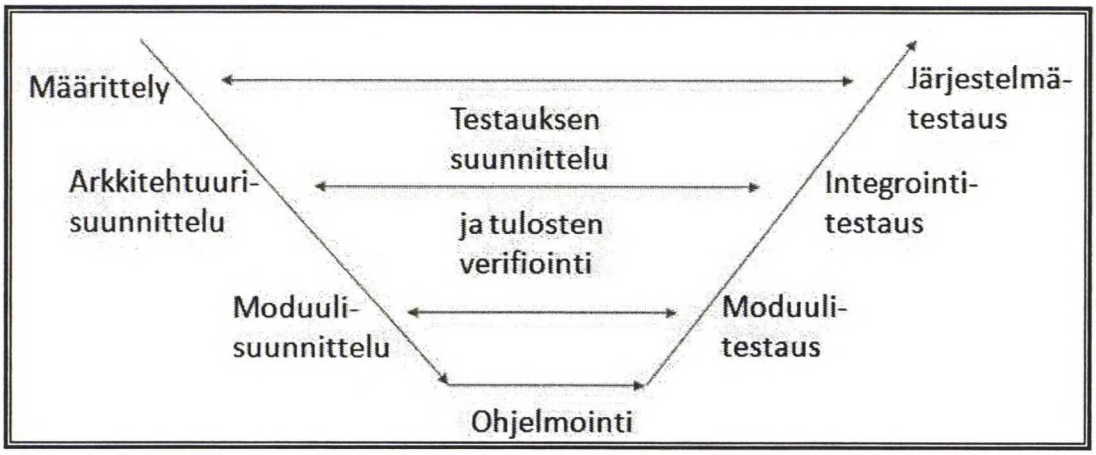
Verkonvalvonta-ohjelmisto käyttävät SMTP sähköpostiprotokollaa (Simple Mail Transfer Protocol) muun muassa vikaviestin välittämiseen sähköposteilla halutuille vastaanottajille. Ongelmana verkonvalvonnassa käytettyjen sähköpostiviestien osalta on se, että vika saattaa esiintyä juuri sähköpostijärjestelmässä tai siihen liittyvissä järjestelmissä. Tällöin sähköpostilla lähetettävää vikaviestiä ei saada toimitettua perille toivottuna hetkenä tai vikaviesti ei välity perille koskaan. Lisäongelmia aiheuttavat myös tilanteet jossa vastaanottajan sähköpostilaatikko on täynnä tai sähköpostien välityksessä esiintyy ongelmia esimerkiksi reititysviasta tai -ongelmista johtuen.

### 3.4.10 HTTPS

Verkonvalvonta-ohjelmistojen selainpohjainen etäkäyttöliittymän tiedonsiirtoon käytetään yleensä tietoturvasyitä johtuen HTTPS (Hypertext Transfer Protocol Secure) protokollaa HTTP protokollan sijaan (Hypertext Transfer Protocol). HTTP protokollan mukaisesti siirrettävää tietoa ei salata vaan se siirretään selkokielisenä käytettävien tietoverkkojen yli. HTTPS protokollan tiedonsiirron salaus voidaan toteuttaa joko SSL (Secure Sockets Layer) tai TLS (Transport Layer Security) tekniikalla, jolloin kaikki etäkäyttöliittymän yli siirrettävä tieto käyttäjätunnistuksesta lähtien salataan [13,18].

# 4 Verkonvalvonta-ohjelmiston ohjelmistokehitys

Verkonvalvonta-ohjelmisto ohjelmoitiin perinteisen V-mallin (Kuva 9) mukaisesti, perustuen määrittelyihin sekä valittuihin arkkitehtuuri- ja moduulisuunnitelmiin [26].




**Kuva 9: Testauksen V-malli**

Ohjelmien testaus määritellään suunnitelmalliseksi virheiden etsimiseksi ohjelmaa tai sen osaa suorittamalla [26]. Ohjelmistojen laajuus ja monimutkaisuus aiheuttavat virheitä, joita kattavalla testaamisella on tarkoitus löytää [33]. Arvioiden mukaan valmiissa ohjelmissa on kattavan testaamisen jälkeen noin 5 prosenttia ohjelmavirheistä jäljellä, näin ollen kattavakaan testaus ei takaa ohjelmiston virheettömyyttä [26]. Testaamisen aikana voidaan löytää suuri joukko virheitä, jotka vaativat korjauksia ohjelmistoon. Testaamisessa suoritettavat testitapaukset yritetään määritellä niin, että suorittamalla se löydetään suurella todennäköisyydellä joku vielä havaitsematon virhe. Testauksen tuloksia verrataan oletettuja tuloksia vastaan. Ohjelmistojen testaamisen kannalta on erittäin tärkeä tietää, mitkä ovat halutut tilat johon ohjelman tulisi päättyä suorituksen loppuessa tai virhetilanteessa. Ilman edellistä tietoa ohjelmisto tulisi testata kaikilla mahdollisilla testitapauksilla, mikä on siihen tarvittavasta ajasta johtuen käytännössä mahdotonta.

Poikkeavan tuloksen esiintyessä testauksen yhteydessä suoritetaan vianetsintä ja korjataan ohjelma. Vianhakuun ja korjauksiin kuluvaa aikaan on vaikea ennakoida ohjelmistoprojekteissa. Mikäli testauksen aikana ei löydetä virheitä, on testaus todennäköisesti epäonnistunut. Syinä virheiden löytymättömyyteen ovat muun muassa: Sovellus suorittaa väärän operaation, palauttaa väärän arvon tai itse testi on väärin laadittu. Testaamiseen kuluvan ajan määräksi on arvioitu olevan yli 30 prosenttia kehitystyöstä. Lisäksi virheiden korjaamiseen tarvittava aika sekä kustannukset nousevat mitä myöhemmin V-mallissa virhe havaitaan (kuva 9). Systeemitestauksessa havaitut virheet voivat viivästyttää ohjelmiston toimitusta tai valmistumista. On myös huomattava, että virheenkorjaus voi aiheuttaa myös uusia virheitä.

Ohjelmistojen monimutkaisuuden hallitsemiseksi niiden osittaminen helpommin hallittaviin kokonaisuuksiin, moduuleihin, on suositeltavaa. Yksi moduuli kuvaa tavallisesti yhden abstraktion tai koostuu joukosta yhteenkuuluvia aliohjelmia. Ohjelmassa käytettäviä globaalit vakiot kootaan yhteen moduuliin, josta ne löytyvät helposti mikäli niiden arvoja tarvitsee muuttaa. Ohjelmiston rakenne ja sen esittäminen perustuu modularisointiin. Ohjelmisto on jaettu erikseen nimettyihin moduuleihin, jotka muodostavat ratkaisun määriteltyyn ongelmaan. Ohjelman jakaminen moduuleihin tekee siitä helpommin hallittavan ja ylläpidettävän, sillä mahdolliset muutokset tehdään tiettyihin moduuleihin, ei koko ohjelmaan.

## 4.1 Moduulitestaus

Moduulitestaus suoritetaan sovelluskehitysvaiheessa ja testattava on yksittäinen moduuli, jossa on yleensä 100–1000 riviä ohjelmakoodia. Moduulitestauksen suorittaa yleensä ohjelmalohkon suunnittelija. Testien perusteella yritetään löytää suurimmat ohjelmisto- ja ohjelmointivirheet ennen integraatiotestausta sekä varmistaa että kyseinen ohjelmalohko täyttää sille asetetut ja vaaditut määrittelyt ja ominaisuudet.

## 4.2 Integraatiotestaus

Integraatiotestauksessa yhdistellään yhteen ohjelmamoduuleja tai ohjelmamoduuliryhmiä. Tarkoituksena on testata moduulien välisten rajapintojen toimivuutta ja varmistaa että testattava ohjelmalohko täyttää sille kehitysvaiheessa asetetut vaatimukset. Integraatiotestaus suoritetaan vasta kun moduulitestaus on suoritettu hyväksytysti. Integraatiotestauksen suorittaa yleensä ohjelmalohkon suunnittelija.

## 4.3 Systeemitestaus

Systeemitestauksen, eli järjestelmätestauksen kohteena on koko järjestelmän toiminnallisuuden testaaminen. Testaaminen keskittyy eniten virheistä toipumiseen, käytettävyyteen, käyttövarmuuteen ja suorituskykyyn liittyviin tekijöihin. Systeemintestaus suoritetaan vasta kun integraatiotestaus on suoritettu hyväksytysti ja testaaminen vastaa yleensä erittäin paljon suunniteltua käyttöympäristöä. Systeemitestauksen suorittaa yleensä kehitystyöstä erillinen systeemitestaaja, jolloin testaamisesta saadaan mahdollisimman objektiiviset tulokset. Systeemitestaukseen voi liittyä myös asiakkaan suorittama hyväksymistestaus.

## 4.4 Regressiotestaus

Regressiotestaus suoritetaan, mikäli ohjelmalohkoa on muutettu. Syitä muuttamiseen voi olla esimerkiksi ohjelmasta löytynyt virhe tai tarve lisätä uusia ominaisuuksia tai toiminallisuutta. Regressiotestaukseen liittyy olennaisesti suuret kulut, sillä muutosten jälkeen suoritetaan aina vähintään systeemitestaus.

## *4.5 Automaatiotestaus*

Automaatiotestaamisella saavutetaan useita etuja manuaalitestaamiseen verrattuna, joita ovat esimerkiksi testauksen nopeutuminen ja toistettavuus sekä inhimillisten virheiden eliminointi.

# 5 Verkonvalvonta-ohjelmisto

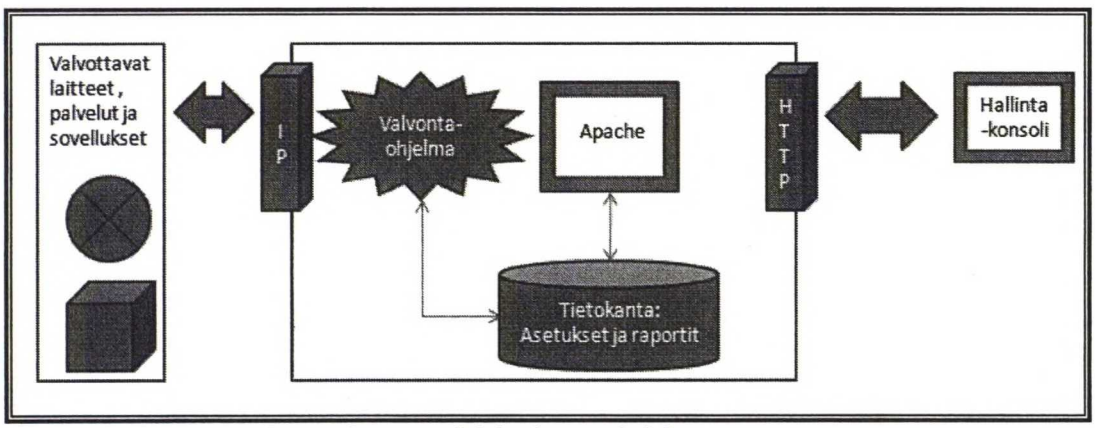
## 5.1 Verkonvalvonta-ohjelmiston vaatimusmäärittely

Verkonvalvonta-ohjelmiston toteutuksen reunaehdoiksi oli määritelty:

- Määriteltyjen laitteiden, sovellusten ja palveluiden toiminnan automaattinen ja periodinen valvonta
- Määriteltyjen laitteiden tietojen automaattinen ja periodinen haku
- Graafinen etäkäyttöliittymä, johon sisältyy käyttäjäntunnistus
- Raportointi tärkeimpien valvontatietojen osalta
- Toteutus käyttäen avoimen lähdekoodin työkaluja ja varusohjelmia
- Lisätoiveena oli havaittujen vikailmoitusten lähettäminen sähköpostitse määritellyille käyttäjille

## 5.2 Verkonvalvonta-ohjelmiston tekninen toteutus

Koska vaatimusmäärittely ei ottanut kantaa toteutuksen arkkitehtuuriratkaisuun, valittiin käytettäväksi arkkitehtuuriratkaisuksi keskitetty verkonvalvonta toteutuksen yksinkertaisuuden vuoksi. Verkonvalvonta-ohjelmisto toteutettiin Perl-ohjelmointikielellä ja sen lisämoduuleilla, Apache palvelinohjelmistolla, RRDtool ja SendMail varusohjelmistoja käyttäen (Kuva 10).




**Kuva 10: Oman verkonvalvonta-ohjelmiston yleiskuvaus**

Määriteltyjen laitteiden, sovellusten ja palveluiden toiminnan automaattinen ja periodinen valvonta sekä tietojen haku valvottavilta laitteilta toteutettiin Perl-ohjelmointikielellä ja sen lisämoduuleilla. Edellä haetuista tiedoista muodostettiin RRDtool-varusohjelmistolla raportit, joista käy ilmi laitteiden ja palveluiden tila sekä laitteiden vasteaika. Apache-palvelinohjelmistolla ja Perl-lisämoduuleilla toteutettiin verkonvalvonta-ohjelmiston graafinen etäkäyttöliittymä toimintoineen sisältäen vaaditun

käyttäjätunnustautumistoiminnon. Lisätoiveena listattu havaittujen vikailmoitusten lähettäminen sähköpostitse määritellyille käyttäjille toteutettiin SendMail-varusohjelmiston avulla. Vaatimusmäärittelyn mukaisesti kaikki toteutuksessa käytetyt ohjelmat ovat avoimen lähdekoodin ohjelmistoja.

### 5.2.1 Perl-ohjelmointikieli ja sen lisämoduulit

Avoimeen lähdekoodiin perustuva Perl-ohjelmointikieli on saanut nimensä kehittäjänsä Larry Wall:n mukaan sanoista Practical Extraction and Report Language [21–22]. Nimensä mukaisesti se on ohjelmointikieli, joka soveltuu erinomaisesti tekstin, tiedostojen ja prosessien muokkaamiseen. Koska Perl suunniteltiin alun perin jonojen käsittelyyn, sisältää se työkalut sekä syöte- ja vasteparametrien muokkaukseen ja suodatukseen että säännönmukaisten lausekkeiden tunnistamiseen ja muokkaukseen. Edellä kuvatut ominaisuudet, Perl:n laitteistoriippumaton alusta ja se, että Perl on saatavilla paljon avoimeen lähdekoodin perustuvia lisämoduuleita tekevät siitä yhden monipuolisimmista ohjelmointiympäristöistä sekä Unix- että Windows ympäristöissä. Ohjelmointikieleen sisältyy piirteitä Unix-työkaluista, kuten Bourne shell:stä. Koska Perl-skriptit voivat käyttää systeemikutsuja ja C-ohjelmointikielen kirjastofunktioita, voidaan se ajatella C-kielen ja shell-skriptien yhdistelmänä.

Perl on ajonaikana kääntävä kieli, joten Perl:llä tehtyjä ohjelmia tai skriptejä ei tarvitse kääntää ennalta erillisellä ohjelmistolla. Perl ohjelmistoista ei myöskään muodosteta erillisiä binääritiedostoja [21–22]. Ajonaikana kääntävien kielien hyvinä puolina on mm. nopeampi ohjelmistokehitys, sillä ohjelmistoja ei ole tarvetta kääntää jokaisen ohjelmistokehitysvaiheen jälkeen. Edellisestä johtuen Perl soveltuu erityisen hyvin prototyyppien suunniteluun, toki myös muihinkin ohjelmistokehityksen vaiheisiin.

Perl:n lisämoduulien avulla saadaan laajennettua toiminallisuutta niissä toteutettujen lisäfunktioiden ja -rajapintojen avulla. Diplomityössäni käytettyyn Perl 5.10 versioon lisämoduulit on asennettu PPM-ohjelmistolla (Perl Package Manager) [30].

#### 5.2.1.1 Perl Net

Perl Net-moduuli sisältää useita tietoverkkokäyttöön tarvittavia rajapintoja, funktiota ja moduuleita. Diplomityössäni verkkomoduulista käytin seuraavia Net-moduuliin kuuluvia Ping- [39] ja SNMP-moduuleita [28].

Kaikille valvonnan piirin kuuluville laitteille suoritetaan yksinkertainen mutta tehokas testaus lähettämällä valvonta-ohjelmistosta TCP/IP Internet Control Message Protocol (ICMP) -protokollan mukainen Echo Request-paketti. Mikäli valvottava laite toimii oikein, se vastaa tähän pakettiin lähettämällä Echo Reply-paketin [1,43]. Ping-moduuli [39] sisältää Ping-funktion valvottavien laitteiden yhteyden toimivuuden testaamiseen. Alla oleva (Esimerkki 2) palauttaa valvonnalle arvon yksi, mikäli valvottu laite vastaa:

```
$p = Net::Ping->new();
$p->hires();
 ($ret, $duration, $ip) = $p->ping($host, 5.5);
if ($ret eq 1)
{
        $rrd_time=($duration*1000);
        @const = split ("\\.", $rrd_time);
        $rrd_time=@const[0];
        $value=1;
}
```

**Esimerkki 2: Valvottavien laitteiden yhteyden testaus Ping-funktiolla**

Järjestelmäaika ja testin tulos, vasteaika millisekunneissa, tallennetaan tiedostoon icmp-IP.txt). Mikäli vastauksen saaminen testattavalta laitteelta ylittää määritellyn raja-arvon, 10 sekuntia, tallennetaan vaste-ajaksi staattinen arvo 10000 kyseiseen laitteen tiedostoon.

SNMP-moduulin avulla valvonta-ohjelmisto hakee valvottavien laitteiden tiedot SNMP protokollaa käyttäen. Käytetty NET::SNMP moduulin toteutus tukee SNMP versiota 1-3 [28]. Alla oleva ohjelma (Esimerkki 3) palauttaa valvonta-ohjelmistolle valvotun laitteen hostname-muuttujan tiedon, mikäli se on määritelty:

```
my ($session, $error) = Net::SNMP->session(-hostname=>$snmp_ip, -
community=>$snmp_communitystring);
$resultsysName = $session->get_request($sysName);
if(defined($resultsysName))
{
        $apuresultsysName= $resultsysName->{$sysName};
        $resultsysName = $session->get_next_request($sysName);}
}
```

**Esimerkki 3: Valvottavan laitteen SNMP-hostname parametrin haku**

Mikäli valvottavalta laitteelta saadaan tarvittavat tiedot, tallennetaan ne SNMP-IP.txt tiedostoon.

### 5.2.1.3 Perl IO

Perl IO-moduuli sisältää useita lisärajapintoja ja funktiota joiden avulla voidaan muun muassa tarkistaa eri sovellusten käyttämien porttien tiloja. Diplomityössäni käytin IO::Socket::INET objektia ja socket-rajapintaa valvottavien sovellusten ja palveluiden toiminnan tarkistamiseen [44].

Alla oleva ohjelma (Esimerkki 4) palauttaa valvonnalle arvon yksi mikäli valvottu palvelu tai sovellus vastaa:

```
$sockettest = IO::Socket::INET->new(PeerAddr=>$hostipaddr,PeerPort
=>$socketport,Proto=>$socketprotocol,Timeout=>$timeout);
if ($sockettest)
{
        $result=1;
}
```

**Esimerkki 4: Valvottavan palvelun tilan tarkistaminen**

Järjestelmäaika ja valvonnan tulos, toimii (1) tai ei-toimi (0), tallennetaan tiedostoon (valvonttavapalvelunnimi-IP.txt). Mikäli vastauksen saaminen testattavalta laitteelta ylittää määritellyn raja-arvon, tallennetaan ei-toimi arvo kyseiseen laitteen palveluvastaus-tiedostoon.

### 5.2.1.4 Perl CGI

Common Gateway Interface (CGI) on standardoitu tekniikka, jota käyttämällä selaimen kautta voidaan välittää tietoja suoritettavalle ohjelmalle. Common Gateway Interface-tekniikan hyvänä puolena on se, että se ei ole sidottu tiettyyn ohjelmointikieleen.

Perl CGI-moduuli tarjoaa edellä kuvatusti keinon luoda HTML-sivuja dynaamisesti sekä tallentaa ja käsitellä käyttäjän syöttämiä tietoja lomakkeista [2, 25].

Alla oleva ohjelma (Esimerkki 5) tallentaa tiedostoonconfig.txt selaimeen syötetyt tekstikentät 1-3 ja parametrit 1-9 sekä suorittaa tämän jälkeen oma.pl ohjelman.

```
#!c:/perl/bin/perl.exe
use strict;
use warnings;
use CGI qw/:standard/;
if (param())
{
        open FILE, '>>', 'C:\confix.txt';
        print FILE param('text1'), ':', param('select1'), ':', param('select2'),
       ':',param('select3'), ':', param('select4'), ':', param('select5'), ':',
param('select6'),       ':',param('select7'),':', param('text2'),':', param('select8'),':',
param('text3'),         ':',param('select9'), "\n";
}
print redirect("oma.pl");
```

**Esimerkki 5: Selaimeen syötettyjen laite ja valvontatietojen lukeminen**

### 5.2.2 Apache

Avoimeen lähdekoodiin perustuva Apache HTTP-palvelinohjelma on erittäin laajalti käytössä, Netcraftin tutkimuksen mukaan elokuussa 2008 puolet kaikista HTTP-palvelimista oli Apache palvelimia [19]. Syynä Apachen suosioon ovat ilmaisen lähdekoodin lisäksi sen laajennettavuus lisämoduulien avulla. Ohjelmistoon sisältyy esimerkiksi mod_cgi, joka mahdollistaa ulkoisten ohjelmien ajamisen CGI-ympäristössä.

Apache-ohjelmistolla toteutettiin verkonvalvonta-ohjelmiston hallintakonsoli, joka on toiminnallisesti etäkäyttöliittymä. Hallintakonsoli sisältää vaatimusmäärittelyn mukaisesti käyttäjätunnistautumisen, joka perustuu ennalta määriteltyihin käyttäjätunnukseen ja salasana pareihin. Onnistuneen sisäänkirjautumisen jälkeen etäkäyttöliittymän kautta on mahdollista lisätä, poistaa sekä muokata valvottavia laitteita ja sovelluksia sekä katsoa valvottavien laitteiden raportointitietoja.

Vaatimusmäärittelyn mukaisen käyttäjätunnistautumisen käyttöönotosta Apachessa (Esimerkki 6) Microsoft Windows ympäristössä [17–18,46-47]:

> Avataan komentotulkki ja siirrytään Apache\Bin-kansioon
> Syötetään komento "htpasswd -c -b passwd.txt kayttajanimi salasana"

**Esimerkki 6: Apache-palvelinohjelman käyttäjätunnistautumisen käyttöönotto**

### 5.2.3 RRDtool

Diplomityössäni käytin avoimeen lähdekoodiin perustuvaa RRDtool-ohjelmistoa (Round Robin Database Tool). Ohjelmisto on laajasti käytetty verkko-operaattorien toimesta muun muassa reitittimien erilaisten porttitilastojen tallentamiseen, analysointiin ja arkistoimiseen [32]. RRDtool:lla muodostin laitteiden, palveluiden ja sovellusten toimintojen vastaustiedostoista tietokannat (Esimerkki 7).

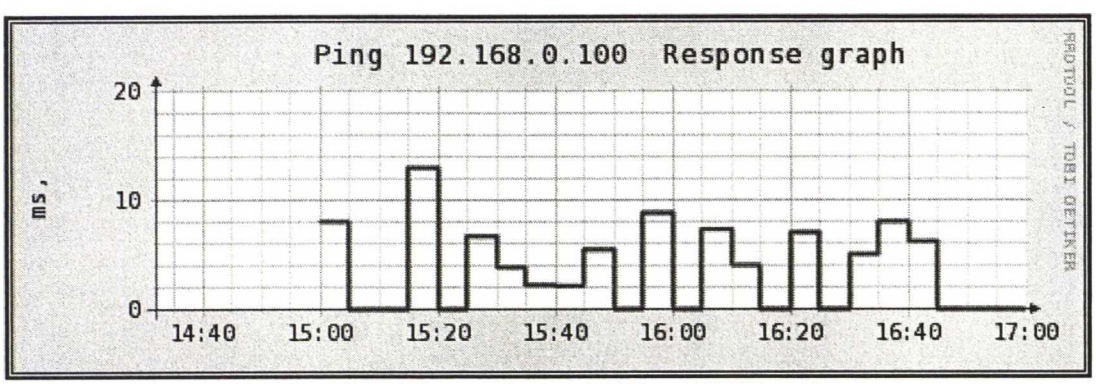
```
my $rrd_db_filename = "$filename".".rrd";
my $rrd_cmd0="rrdtool";
my $rrd_cmd1="create";
my $rrd_cmd2="start";
.....
@args=("$rrd_cmd0 $rrd_cmd1 $rrd_db_filename --$rrd_cmd2 $starttime
DS:response:GAUGE:300:0:10000 RRA:LAST:0.5:1:24
RRA:AVERAGE:0.5:6:10");
.....
```

**Esimerkki 7: Tietokannan muodostaminen RRDtool-ohjelmistolla**

Tietokannoista muodostettiin RRDtool-ohjelmistolla graafisia kuvaajia (Kuva 11) muun muassa valvottavien laitteiden vasteaikojen osalta [2-5] (Esimerkki 8).

```
...
my $rrd_cmd0="rrdtool";
my $rrd_cmd2="start";
my $rrd_cmd6="graph";
.....
@args = ("$rrd_cmd0 $rrd_cmd6 $rrd_pic_filename --$rrd_cmd2 $starttime --end
$endtime --title $title $test $hostipaddr $title2 $title3 --vertical-label ms,
DEF:myresponse=$rrd_db_filename:response:LAST
LINE2:myresponse#FF0000");
system (@args)==0 or die "Cannot create RRD-picture  $rrd_db_filename:
system @args failed\n";
```

**Esimerkki 8: Graafisen kuvaajan muodostaminen RRDtool-ohjelmistolla**




**Kuva 11: Valvottavan verkkolaitteen kuvaaja Ping/ICMP-valvonnan osalta**

Nimensä mukaisesti RRDtool-ohjelmisto kirjoittaa tietokannan vanhimman tiedon päälle, mikäli tietokanta alkaa täyttyä. Tällä varmistetaan että tietokantojen koko ei kasva liikaa, eikä tietojen tallennus johda ongelmiin levytilan täyttymisen muodossa. Tietokantaa tallennetaan oletuksena tietoja ennalta määritellyin väliajoin, ja mikäli tietoa ei tänä ajankohtana ole saatavissa ei kuvaajia saada muodostettua oikein. Edellisestä johtuen verkonvalvonta-ohjelmistojen suorittamien hakujen synkronointi on erittäin tärkeää. Työssä käytetty RRDtool versio oli 1.2.28 [7]. Kattavampi esimerkki RRDtool-ohjelmiston käytöstä on liitteessä 1.

### 5.2.4 SendMail

Lisätoiveena listattu havaittujen vikailmoitusten lähettäminen sähköpostitse määritellyille käyttäjille toteutettiin SendMail-varusohjelmiston avulla. Sähköpostiviesti lähetetään (Esimerkki 9) mikäli valvottavassa laitteessa, sovelluksessa tai palvelussa havaitaan

vikatilanne ja sähköpostitoiminto on määritelty käyttöön kyseisen laitteen osalta. Diplomityössä käytetyn SendMail-ohjelmiston versio on 1.2 [28].

```
if ($result != 1)
{
        $mailsubject= "Error in $testedhost";
        $mailtext="$test FAILED FROM $omaip TO $testedhost";
        @args=("sendmail","/smtpsrv","$mailsrv","/to","$mailsender","/from",
        "$mailrecipient","/subject", $mailsubject, "/body", $mailtext);
        system (@args) ==0 or die "system @args failed\n";
}
```

**Esimerkki 9: Sähköpostiviestin muodostaminen ja lähetystoiminne**

## 5.3 Verkonvalvonta-ohjelmiston toiminnallisuuden kuvaus

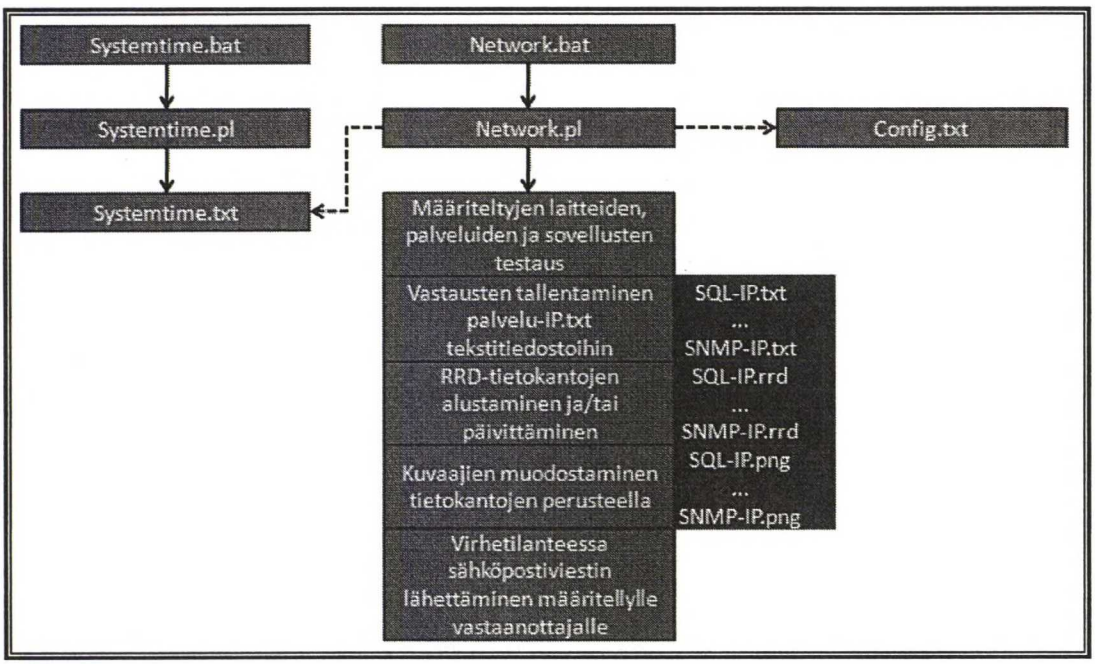
Loogistesti verkonvalvonta-ohjelmiston toiminnallisuus koostuu kahdesta osasta: verkonvalvonta-ohjelmasta sekä sen etäkäyttöliittymästä. Liitteessä kaksi on listattu verkonvalvonta-ohjelmiston käyttämät hakemistot sekä ohjelmat

### 5.3.1 Verkonvalvonta-ohjelmiston toiminnallinen kuvaus

Verkonvalvonta-ohjelmiston toiminta (Kuva 12) perustuu kahden ajastetun skriptin suorittamisen, jotka suoritetaan periodisesti. Diplomityössäni skriptit käynnistetään joka viides minuutti.

Systemtime.bat, käynnistää systemtime.pl-skriptin, joka tallentaa järjestelmäajan Epoch-muodossa systemtime tekstitiedostoon. RRDtool vaatii toimiakseen, että järjestelmäaikana käytetään Epoch-muotoa, kuluneiden sekuntien määrä 1.1.1970 UTC lähtien [4-6,23].

Network.bat skripti käynnistää network.pl verkonvalvonta-ohjelmiston suorittamisen. Ohjelmisto lukee config.txt tiedostosta valvottavat laitteet, sovellukset ja palvelut sekä näihin liittyvät lisäparametrit kuten SNMP-yhteisönimen. Samassa tiedostossa on myös määritelty lähetetäänkö vikatilanteessa sähköpostiviestejä sekä osoitteen, johon nämä viestit lähetetään.

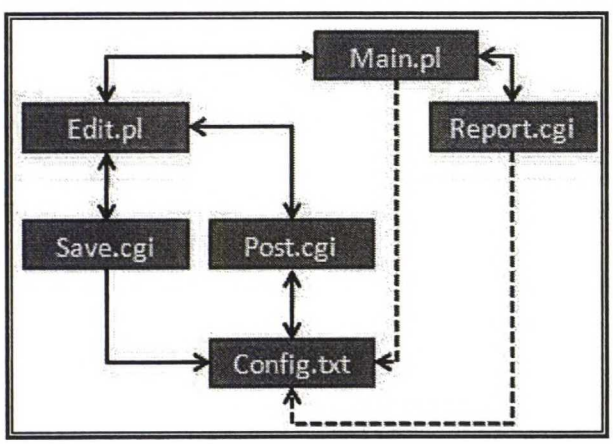



**Kuva 12: Verkonvalvonta-ohjelmiston toiminnallinen kuvaus**

Tämän jälkeen suoritetaan valvontatoimet ja talletetaan niihin saadut vastaukset tekstitiedostoihin. Vastaustekstitiedostojen perusteella luodaan ja/tai päivitetään tietokannat sekä muodostetaan niiden avulla graafiset kuvaajat valvottavista laitteista, palveluista ja sovelluksista vasteajoista ja toiminnasta. Mikäli vikatilanteista on määritelty sähköpostiviestien vastaanottaja, lähetetään havaituista vikatilanteista virheilmoitus tähän osoitteeseen.

### 5.3.2 Etäkäyttöliittymän toiminnallinen kuvaus

Verkonvalvonta-ohjelmiston etäkäyttöliittymä on toteutettu Perl-ohjelmilla ja CGI-skripteillä, joiden toiminnallisuus kuvataan alla ohjelmalohkoittain (Kuva 13).



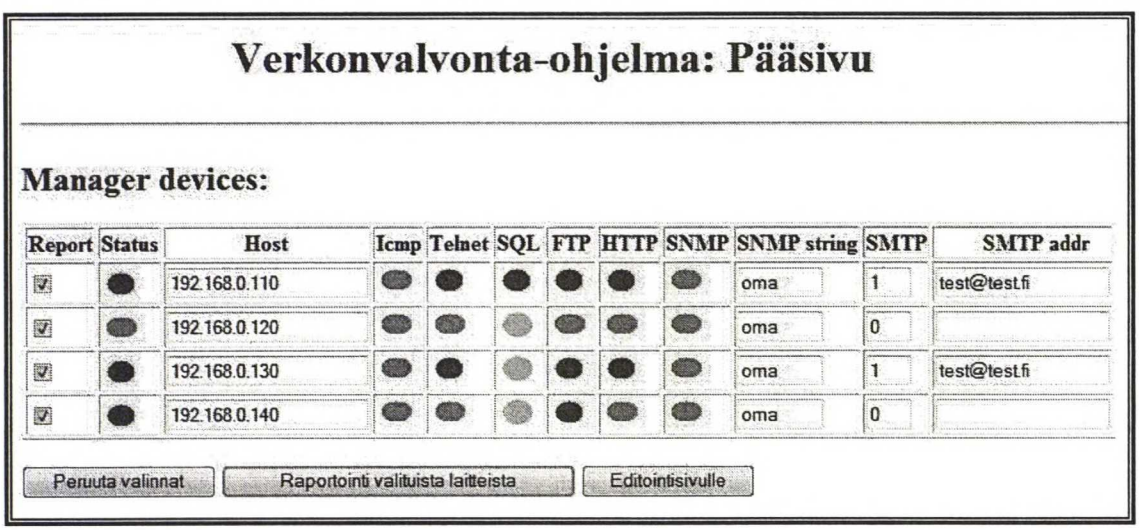
**Kuva 13: Verkonvalvonta-ohjelmiston etäkäyttöliittymän toiminnallinen kuvaus**

### 5.3.2.1 Main.pl

Verkonvalvonta-ohjelman pääsivu, mail.pl lukee config.txt tekstitiedostoista mitkä laitteet, palvelut ja sovellukset kuuluvat valvonnan piiriin. Tämän jälkeen luetaan vastaavien laitteiden ja palveluiden vastaustiedostot, joiden tilatieto kuvataan pääsivulla värisymboleilla yleistasolla sekä valvontaspesifisesti (Kuva 14):

- Vihreä: Toimii
- Punainen: Ei toimi
- Harmaa: Palvelua tai sovellusta ei ole määritelty valvonnan piiriin

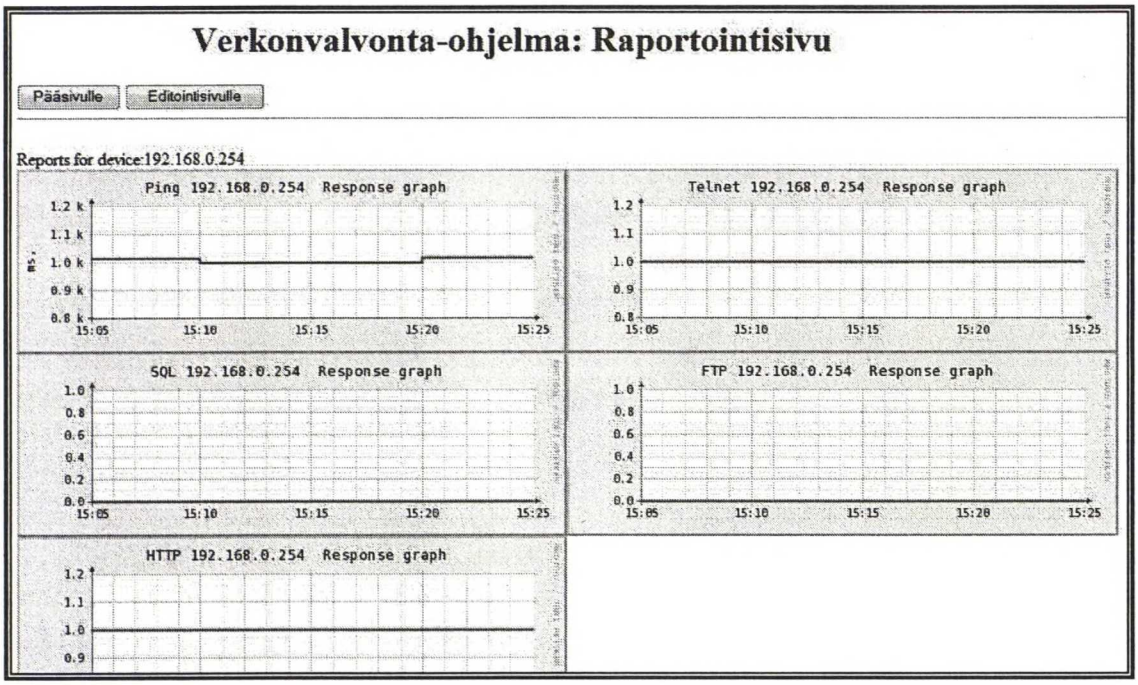
Pääsivulta voidaan siirtyä editointi - ja raportointisivulle.




**Kuva 14: Verkonvalvonta-ohjelmiston Etäkäyttöliittymän pääsivu**

### 5.3.2.2 Report.cgi

Raportointisivustolla näytetään raportoitavien laitteiden, palveluiden ja sovellusten raportointitiedot kuvina sekä SNMP-tiedot tekstinä. Sivulta voidaan siirtyä pää- ja editointisivuille (Kuva 15).




**Kuva 15: Verkonvalvonta-ohjelmiston etäkäyttöliittymän Raportointisivu**

### 5.3.2.3 Edit.pl

Editointisivulla voidaan lisätä, poistaa ja muokata valvottavia laitteita, palveluita ja sovelluksia save.cgi ja post.cgi skriptien avulla (Kuva 16). Editointisivulta voidaan siirtyä pääsivulle.


**Kuva 16: Verkonvalvonta-ohjelmiston Editointisivu**

#### 5.3.2.4 Save.cgi

Save.cgi-skripti muokkaa config.txt tekstitiedostoa editointisivulle tehtyjen muutosten, poistojen tai lisäysten mukaisesti sekä päivittää editointisivun tiedot dynaamisesti tallennuksen jälkeen.

#### 5.3.2.5 Post.cgi

Kun laitteita lisätään Editointi sivulla, post.cgi-skripti lukee ja tallentaa uuden laitteen tiedot config.txt tekstitiedoston sekä päivittää muutokset dynaamisesti editointi-sivun laitelistalle.

# 6 Analyysi

Oman verkonvalvonta-ohjelmiston toiminnan kattavuuden testaamiseksi

- Valvottavien laitteiden ja palveluiden satunnainen pysäyttäminen/käynnistäminen sekä valvottavien tilatietojen muutosten verifiointi hallinta- ja raportointinäkymistä. Lisäksi sähköpostihälytysten toiminnan verifiointi vikatilanteiden jatkoraportoinnin osalta
- Käyttäjätunnistautumisen testaaminen väärillä ja oikeilla tunnuksilla ja salasanoilla
- SNMP-tietojen testaaminen väärillä ja oikeilla lähtöarvoilla
- Käyttöliittymän testaaminen
    - Valvottavien laitteiden lisääminen ja poistaminen
    - Valvottavien laitteiden parametrien muuttaminen
- Ohjelman ajastetun skriptin toiminnan testaaminen, jolloin ohjelma käynnistyy ja sitä suoritetaan automaattisesti taustalla ilman käyttäjätoimia

## 6.1 Valvonta-ohjelmiston testiympäristö

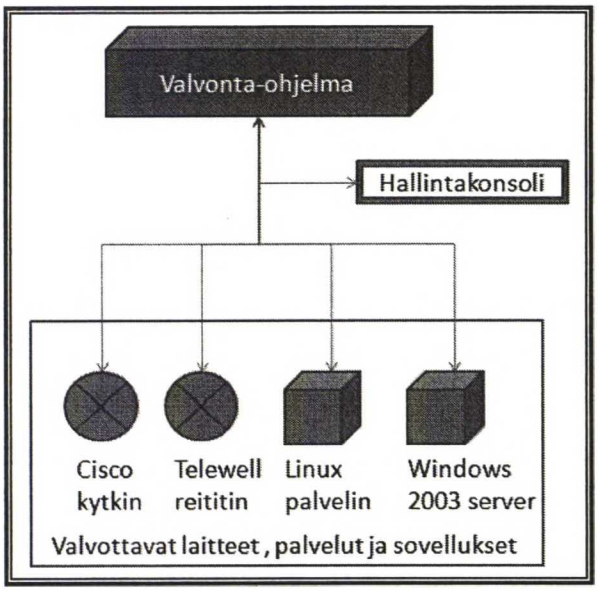
Verkonvalvonta-ohjelmisto tarvittavine ohjelmistoineen (ks. kpl 5.2) asennettiin tietokoneelle, jossa käyttöjärjestelmänä oli Microsoft Windows Vista Ultimate 64-bit versio.

Verkonvalvonta-ohjelmiston toiminnan verifiointia varten rakennettiin testiympäristö, joka koostui sekä fyysisistä että virtuaalilaitteistoista (Kuva 17). Virtuaaliympäristön toteuttaminen helpotti Windows- ja Unix-käyttöjärjestelmien nopeaa testausta omassa eristetyssä ympäristössä. Valvottavat palvelut asetuksineen määriteltiin jokaiseen testilaiteeseen seuraavasti.

- Palvelimissa palveluista oli päällä SQL, FTP, TELNET, HTTP ja SNMP
- Kytkimissä (2kpl) päällä oli FTP, TELNET, HTTP ja SNMP
- Reitittimessä päällä oli FTP, TELNET, HTTP ja SNMP

Edellä kuvatut palvelut käyttivät standardiportteja, jotta testaus vastaisi mahdollisimman hyvin tuotantoympäristöä.

SNMP agenttiin asennettiin ja kaikkiin valvottaviin laitteisiin, ja kaikkiin laitteisiin määriteltiin sama SNMP yhteisönimi. SNMP-osalta määriteltiin valvonta-ohjelmistolle lukuoikeudet kaikkien laitteiden tietoihin. Lisäksi SNMP Trap sanomien vastaanottajaksi määriteltiin valvonta-ohjelmiston IP-osoite.




**Kuva 17: Verkonvalvonta-ohjelmiston testiympäristö**

## 6.2 Testeissä saadut tulokset

Diplomityössäni en käyttänyt testaamisessa työkaluja, joilla olisi voinut arvioida testaamisen eri vaiheiden prosentuaalista kattavuutta, koska testaukset suoritettiin manuaalitestauksena. Manuaalitestaus oli mahdollista, koska kirjoitetun koodin määrä oli varsin pieni, alle 10 000 riviä, ja yllä kuvattujen testivaiheiden suorittaminen oli nopeaa sekä testitapausten määrä oli hyvin rajattu, alle 100 kappaletta. Regressiotestauksen tulokset on sisällytetty alla oleviin kappaleisiin, sillä muutosten yhteydessä olen suorittanut kaikki testausvaiheet alusta lähtien uudelleen.

### 6.2.1 Moduulitestauksen tulokset

Moduulitestausvaiheessa ohjelmistovirheitä löytyi erittäin paljon ja niiden korjaaminen sekä uudelleen testaus oli erittäin nopeaa.

Diplomityössäni testasin kaikki oletetut testitapaukset moduulitestausvaiheessa kuten alla kuvatun testitapauksen (Esimerkki 10).

- Testitapaus 1: Uuden valvottavan laitteen lisääminen valittujen valvontaparametrien kanssa komentoriviltä:
  - Annettava syöte: "10.20.30.40:1:1:1:1:1::::1"
- Onnistunut tulos: Valittujen tietojen tallennus määriteltyyn tekstitiedostoon määrämuodossa yhdelle riville.
  - 10.20.30.40:1:1:1:1:1::::1

- Hyväksytty vikatilanne: Mikäli yllä oleva ei onnistu, muutoksia ei tallenneta tekstitiedostoon ja komentorivillä näkyy virheilmoitus.

**Esimerkki 10: Moduulitestauksen testitapaus**

### 6.2.2 Integraatiotestauksen tulokset

Diplomityössäni testasin kaikki oletetut testitapaukset integraatiotestausvaiheessa, kuten alla kuvatun testitapauksen (Esimerkki 11).

- Testitapaus 1: Uuden valvottavan laitteen lisääminen valittujen valvontaparametrien kanssa web-käyttöliittymän editointisivun kautta:
  - Host:10.20.30.40/ICMP:On/Telnet:On/SQL:On/FTP:On/http:On/Nslooku p:On/SNMP: Off/Snmp community name:/Smtp address:/Hosttype: server
- Onnistunut tulos: Valittujen tietojen tallennus määriteltyyn tekstitiedostoon määrämuodossa
  - 10.20.30.40:1:1:1:1:1::::1
- Sallittu vikatilanne: Mikäli yllä oleva ei onnistu, muutoksia ei tallenneta tekstitiedostoon.

**Esimerkki 11: Integraatiotestauksen testitapaus**

### 6.2.3 Systeemi- ja hyväksymistestauksen tulokset

Diplomityössäni suoritin systeemitestauksen ja hyväksymistestauksen, kuten alla kuvatun testitapauksen (Esimerkki 12).

- Testitapaus 1: Sisään kirjautumien verkonvalvonta-ohjelmistoon ja siirtyminen pääsivulta editointisivulle ja uuden valvottavan laitteiden lisääminen valittujen valvontaparametrien kanssa:
  - Host:192.1680.0.140/ICMP:On/Telnet:On/SQL:On/FTP:On/http:On/Nslo okup:On/SNMP: Off/Snmp community name:/Smtp address:/Hosttype: server
- Onnistunut tulos:
  - Käyttäjän tunnistaminen käyttäjänimellä ja salasanalla.
  - Onnistunut siirtymä Pääsivulta Editointisivulle.
  - Uuden laiteen lisäys ja valittujen tietojen tallennus määriteltyyn tekstitiedostoon määrämuodossa: 192.168.0.140:1:1:1:1:1::::1
- Sallittu vikatilanne:
  - Uuden laitteen lisäämisen osalta: Mikäli lisäys ei onnistu täysmääräisesti, muutoksia ei tallenneta tekstitiedostoon.

**Esimerkki 12: Systeemitestauksen testitapaus**

## 6.3 Testitulosten luotettavuus

Testien luotettavuuden osalta suuri vaikuttava tekijä oli määrittelydokumentaatio. Sitä täydennettäessä ja päivittämällä myös testitapaukset saatiin määriteltyä oikein. Ilman määrittelydokumentaatiota kattava ja luotettava testaaminen on mahdotonta, kuten diplomityön alkuvaiheessa huomasin.

## 6.4 Etäkäyttöliittymä

Etäkäyttöliittymä ja siihen liittyvien teknisten asioiden, kuten käyttäjien tunnistus, onnistui hyvin. Apache WWW-palvelimen asennus ja konfigurointi oli suoraviivaista. Käyttäjä kirjautuu verkonvalvonta-ohjelmistoon Internet-selaimella käyttäen salaamatonta HTTP-yhteyttä. Tämän jälkeen käyttäjä sisään kirjautuu käyttäjätunnuksella ja salasanalla verkonvalvonta-ohjelmistoon.

Käyttöliittymäsuunnittelun aikana tarvittavien valvonta-ohjelmiston sivujen määrä minimoitiin, jotta valvonta-ohjelmiston käyttö oli mahdollisimman tehokasta ja yksinkertaista [26]. Kaikki turha toiminnallisuus on poistettu loppukäyttäjälle näkyvästä graafisesta käyttöliittymästä. Valvonnan osalta kaikki kriittiset tilatiedot esitetään värisymboleilla, jotta valvottavissa laitteissa ja/tai järjestelmissä mahdollisesti esiintyvät virhetilanteet havaitaan käyttöliittymästä mahdollisimman nopeasti.

## 6.5 Raportointi

Tärkeimpien valvontatietojen raportoinnin tekninen toteutus onnistui, eli valvonta-ohjelmiston raportointisivu tuottaa tarvittavat tiedot valvottavien laitteiden, palveluiden ja toimintojen osalta (Kuva 15). Verkonvalvonnan osalta raportointia varten voitaisiin kerätä laajempaa tietoa järjestelmien toimivuudesta tarkempaa analysointia varten. Raportoinnissa käytetään usein tietokanta-ohjelmistoa mutta diplomityössä käytettiin raportoinnin osalta vain erillisiä tekstitiedostoja raporttien tallentamista, kirjoittamista ja analysointia varten.

## 6.6 Vika-ilmoitukset

Vikailmoitusten osalta vaatimusmäärittelyssä lisätoiveena listattu sähköposti-ilmoitus toteutettiin Sendmail-varusohjelmalla.

# 7 Arviointi

Oman verkonvalvonta-ohjelmiston sekä vertailtavan HP SIM-ohjelmiston asennus, testaus ja analysointi suoritettiin laitteistossa, jossa käyttöjärjestelmänä oli Microsoft Windows Vista Ultimate 64-bit versio. Valvottavien laitteiden ja palveluiden osalta testiympäristö on kuvattu tarkemmin kappaleessa 6.1. NetEye ohjelmiston osalta vertailu ja arviointi suoritettiin perustuen sekä tuotteen dokumentaatioon, demo-ohjelmistoon [13–14,45] että omiin käyttökokemuksiin tuotteesta.

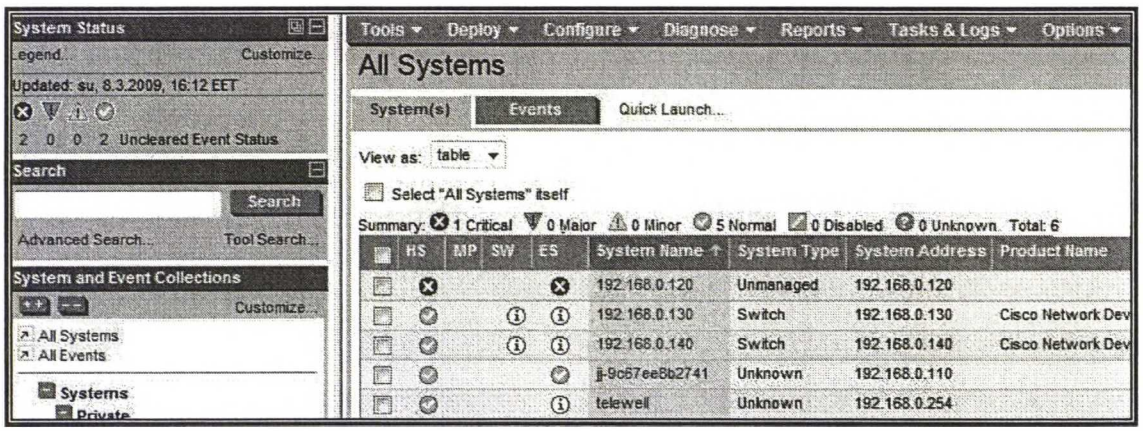
Vertailuun käytetty HP SIM:n versio oli 5.2 SP2 [7] ja NetEye version oli 3.4.9 [14].

## *7.1 HP SIM*

HP SIM on ilmainen ohjelmistopohjainen laitetason valvonta- ja hallinta-ohjelmisto. Sen avulla voidaan valvoa muun muassa verkkoon kytkettyjen reitittimien, kytkimien sekä palvelimien tilaa. Arkkitehtuuriltaan SIM on keskitetty verkonvalvonta-ratkaisu, mutta toiminnallisuudeltaan se tukee myös hierarkista verkonvalvonta-arkkitehtuuria. Tällöin se toimii alemman tason verkonvalvonta-ohjelmistona välittäen saamansa tiedot ylätason verkonvalvonta-ohjelmalle, kuten HP Open View:lle. Valmistajan Microsoft ympäristöön tekemä ohjelmiston asennuspaketti sisältää kaikki tarvittavat ohjelmistot mukaan lukien SQL 2005 Express tietokannan sekä TomCat WWW-palvelinohjelmiston [7]. Mikäli valvottavassa ympäristössä on jo käytössä Oracle-, MSDE- tai SQL-tietokanta, tukee HP SIM ohjelmisto myös näitä. Loogisesti tietokannat voivat sijaita joko samalla tai eri palvelimella kuin itse valvonta-ohjelmisto. HP:n SIM ohjelmistosta on saatavilla asennuspaketti myös Linux ja Unix käyttöjärjestelmiin, mutta näiden asennus, eikä toimintojen vertailu sisältynyt diplomityöhön [10,12]. Verrattuna oman verkonvalvonta-ohjelmiston asennukseen, oli HP SIM ohjelmiston asennus suoraviivaista ja nopeaa.

Koska HP SIM käyttää verkonvalvontaan standardiprotokollia, kuten SNMPv2 ja ICMP, voidaan valvonta- ja raportointitietoja kerätä automaattisesti eri valmistajien laitteista ja palveluista sekä halutuista sovelluksista ja prosesseista. Ohjelmisto käyttää SNMP-protokollan haku-operaatiota valvontatietojen hakuun, muun muassa palvelimilla käynnissä olevista prosesseista [10,12]. Valmistajan omista laitteista, kuten palvelimista, voidaan lukea komponenttitason parametrien arvoja. Luettujen tietojen perusteella saadaan ennakoivaa tietoa mahdollisista laitteistovioista [10,12]

SIM-ohjelmistoa käytetään graafisen käyttöliittymän kautta. Graafisen käyttöliittymän aloitussivulla on yleisnäkymä valvottavien laitteiden toimivuudesta sekä tiedot valvonta-ohjelmistolle välitetyistä vika- ja virheviesteistä (kuva 18). Valvottavien palveluiden, laitteiden ja sovellusten tila kuvataan käyttöliittymässä liikennevalojen mukaisilla värisymboleilla sekä tekstimuodossa (Kuvat 18–19).




**Kuva 18: HP SIM ohjelmiston valvontanäkymä**

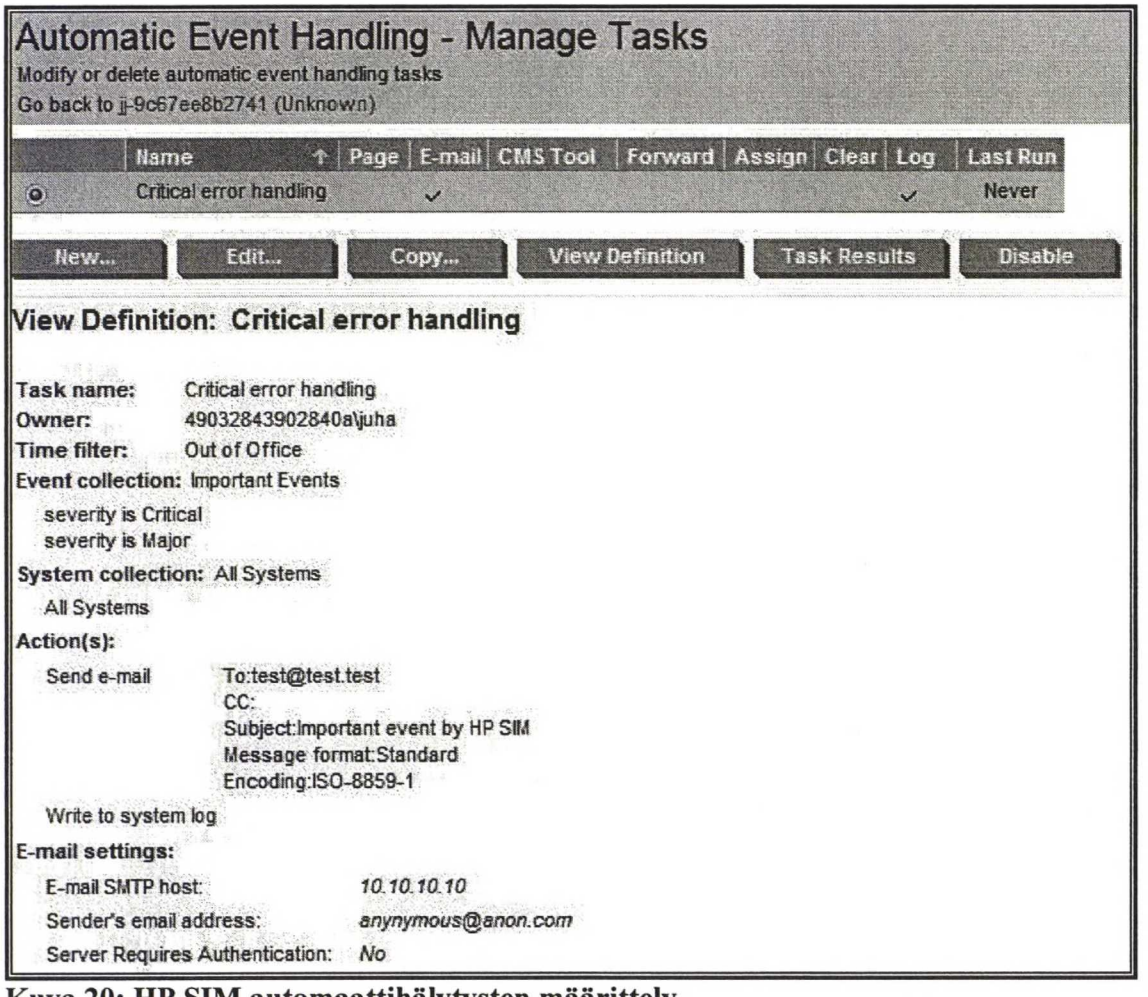
| System Name | Severity | Type | Cleared Status | Received Time | Description |
|---|---|---|---|---|---|
| 192.168.0.100 | Critical | System is unreachable | Cleared | Fri, 11/14/2008, 8:16 PM EET | The current system is no longer reachable |
| 192.168.0.100 | Critical | System is unreachable | Cleared | Fri, 11/14/2008, 9:36 PM EET | The current system is no longer reachable |
| 192.168.0.120 | Critical | System is unreachable | Not cleared | Fri, 2/6/2009, 12:15 PM EET | The current system is no longer reachable |
| 192.168.0.130 | Critical | System is unreachable | Not cleared | Sun, 3/8/2009, 4:22 PM EET | The current system is no longer reachable |
| 192.168.0.140 | Critical | System is unreachable | Not cleared | Sun, 3/8/2009, 4:22 PM EET | The current system is no longer reachable |

**Kuva 19: HP SIM valvontanäkymä toimimattomista laitteista**

Laitteiden lisääminen valvonnan piiriin on yksinkertaista, sillä ohjelmisto sisältää erillisen First Time Wizard-aputyökalun [10]. Työkalun avulla voidaan määritellä verkko-avaruus, jonka laitteet halutaan liittää automaattisen valvonnan piiriin poislukien erikseen määritellyt poikkeukset. Vastaava toimenpide on myös mahdollista suorittaa laitekohtaisesti, mikäli näin halutaan. Laitteiden toimivuuden tarkistus suoritetaan periodisilla ICMP Ping-komennoilla, joiden vastausten perusteella valvottavan laitteen tila kuvataan käyttöliittymässä liikennevalojen mukaisilla värisymboleilla (kuva 18). Erilaisia toistuvia ajastettuja tehtäviä on helppo luoda valmiiden mallitehtävien avulla. Ajastettujen tehtävien avulla voidaan luoda tarvittavat valvontatehtävät itse.

SIM kategorisoi automaattisesti valvottavat laitteet erilaisiin ryhmiin muun muassa käyttöjärjestelmän ja laitteiston valmistajan mukaisesti, mutta käyttäjäkohtainen kustomointi on myös mahdollista.

SIM:n voidaan määritellä erilaisia automaattitoimintoja, mikäli ennalta määritelty tila havaitaan esimerkiksi valvottavassa laitteessa. Kuvassa 20 automaattitoimintoinnoksi on määritelty vikaviestin välitys, mikäli valvottavassa järjestelmässä havaitaan kriittinen tai vakava vika. Tieto voidaan lähettää joko teksti- tai sähköpostiviesteillä erilaisten ennalta määriteltyjen sääntöjen mukaisesti tai kirjoittaa se valvonta-järjestelmän järjestelmälokitiedostoon. Tietojen lähettämistä voidaan rajoittaa vastaanottajan, viikonpäivän ja kellonajan mukaan, esimerkiksi lähettämällä tieto tietystä viasta ainoastaan normaalin työajan ulkopuolella tietylle henkilölle tekstiviestinä.



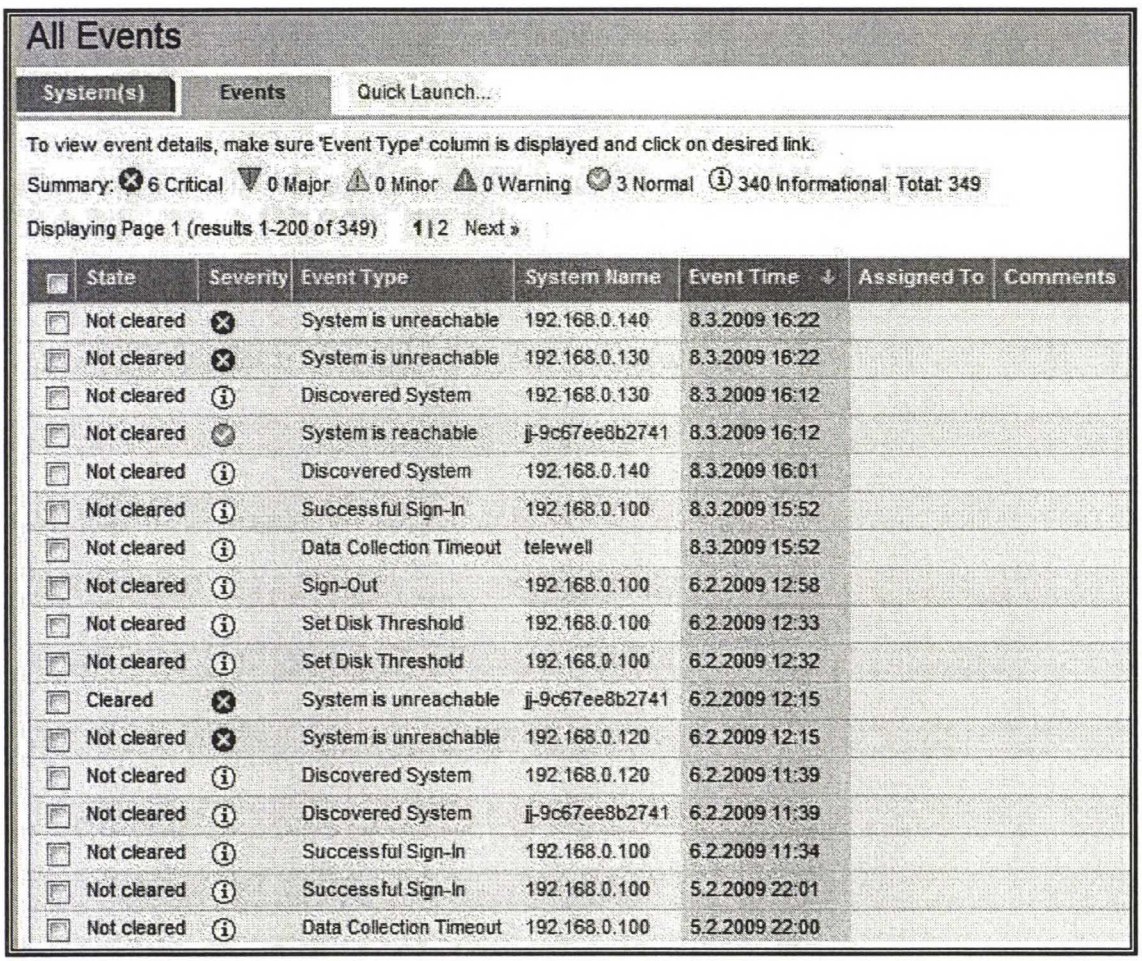

**Kuva 20: HP SIM automaattihälytysten määrittely**

Tietoturvaan liittyvät asiat on huomioitu kattavasti: Valvonta-ohjelmiston toteutus sisältää roolipohjaisen käyttäjäryhmittelyn, jonka avulla on mahdollista rajoittaa järjestelmän käyttöä erilaisten ryhmäkohtaisten tarpeiden mukaisesti tavallisen käyttäjätason määrittelyn lisäksi. Ohjelmiston Windows ja Unix versiot tukevat käyttöjärjestelmätason käyttäjätunnistusmekanismeja (Win32 ja PAM), jolloin valvonta-ohjelmiston käyttöä varten ei tarvita välttämättä erillistunnuksia [12]. Tietoliikenne selainpohjaisen käyttöliittymän ja hallinta-ohjelmiston välillä on salattu SSL tekniikkaa käyttäen [10,12].

SIM ohjelmistoon sisältyy verkonvalvonnassa useasti tarvittavia aputyökaluja, kuten Ping-komento. Sen avulla voidaan tarkistaa nopeasti valvottavien laitteiden toimivuus.

Raportoinnin osalta tuote sisältää kattavan valikoiman valmiita raporttimalleja (Kuva 21) sekä mahdollisuuden luoda omia raportteja. Raporttimallien avulla palveluita valvova

henkilö voi muodostaa nopeasti esimerkiksi raportin laitteista, joissa on esiintynyt kriittiseksi luokiteltuja vikoja.



All Events

System(s) | Events | Quick Launch...

To view event details, make sure 'Event Type' column is displayed and click on desired link.

Summary: 6 Critical, 0 Major, 0 Minor, 0 Warning, 3 Normal, 340 Informational, Total: 349

Displaying Page 1 (results 1-200 of 349) 1 | 2 Next »

| State | Severity | Event Type | System Name | Event Time | Assigned To | Comments |
|---|---|---|---|---|---|---|
| Not cleared | Critical | System is unreachable | 192.168.0.140 | 8.3.2009 16:22 | | |
| Not cleared | Critical | System is unreachable | 192.168.0.130 | 8.3.2009 16:22 | | |
| Not cleared | Informational | Discovered System | 192.168.0.130 | 8.3.2009 16:12 | | |
| Not cleared | Normal | System is reachable | jj-9c67ee8b2741 | 8.3.2009 16:12 | | |
| Not cleared | Informational | Discovered System | 192.168.0.140 | 8.3.2009 16:01 | | |
| Not cleared | Informational | Successful Sign-In | 192.168.0.100 | 8.3.2009 15:52 | | |
| Not cleared | Informational | Data Collection Timeout | telewell | 8.3.2009 15:52 | | |
| Not cleared | Informational | Sign-Out | 192.168.0.100 | 6.2.2009 12:58 | | |
| Not cleared | Informational | Set Disk Threshold | 192.168.0.100 | 6.2.2009 12:33 | | |
| Not cleared | Informational | Set Disk Threshold | 192.168.0.100 | 6.2.2009 12:32 | | |
| Cleared | Critical | System is unreachable | jj-9c67ee8b2741 | 6.2.2009 12:15 | | |
| Not cleared | Critical | System is unreachable | 192.168.0.120 | 6.2.2009 12:15 | | |
| Not cleared | Informational | Discovered System | 192.168.0.120 | 6.2.2009 11:39 | | |
| Not cleared | Informational | Discovered System | jj-9c67ee8b2741 | 6.2.2009 11:39 | | |
| Not cleared | Informational | Successful Sign-In | 192.168.0.100 | 6.2.2009 11:34 | | |
| Not cleared | Informational | Successful Sign-In | 192.168.0.100 | 5.2.2009 22:01 | | |
| Not cleared | Informational | Data Collection Timeout | 192.168.0.100 | 5.2.2009 22:00 | | |

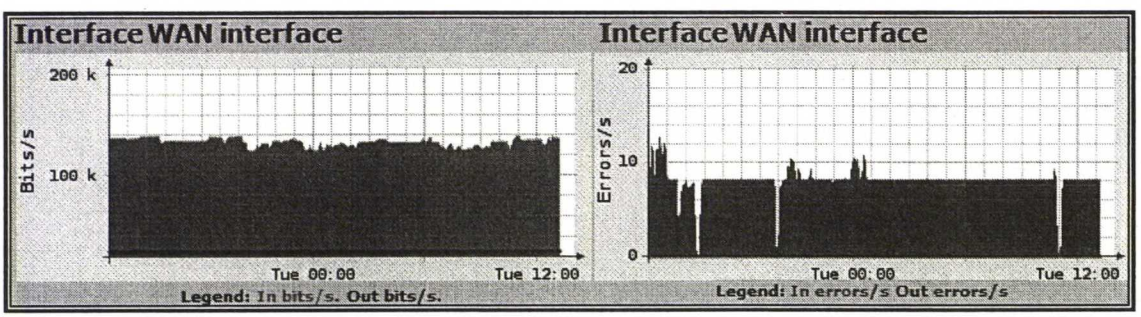
**Kuva 21: HP SIM raportti**

SIM:n hallintatoimintojen avulla on mahdollista asentaa valmistajan omiin laitteisiin, kuten palvelimiin, esimerkiksi BIOS- (Basic Input Output System) ja laitteistoajuripäivityksiä. SIM:n kautta voidaan suorittaa ajuripäivityksiä HP:n omille laitteille. Päivitykset haetaan ja tallennetaan SIM ohjelmiston tietokantaan, josta ne voidaan päivittää keskitetysti halutuille valvottaville ja hallittaville laitteille.

Hallintatoimintojen kautta on myös mahdollista muodostaa salattu etäyhteys valvottavaan tai hallittavaan laitteeseen. Ohjelmaan saatavien lisäkomponenttien avulla voidaan valvoa, raportoida, etähallita ja päivittää laitteistovalmistajan omia tuotteita. Laajemman hallinnan osalta hallittaviin laitteistoihin on asennettava laitteistovalmistajan kehittämä agentti-ohjelmisto, jonka avulla lisätoimitteet voidaan suorittaa. Lisä-agenttien avulla laitteet voivat lähettää SNMP Trap viestejä esimerkiksi erilaisten laitekomponenttien lämpötilaraja-arvon ylittymisestä. Viestien näytetään hallinta-ohjelmiston käyttöliittymän etusivulla varoituksina. Valvonta-ohjelmisto kautta nähdään todennäköistesti vikaantuvat laitteiston komponentit, jotka voidaan korvata ennen kuin virhe tai vika realisoituu.

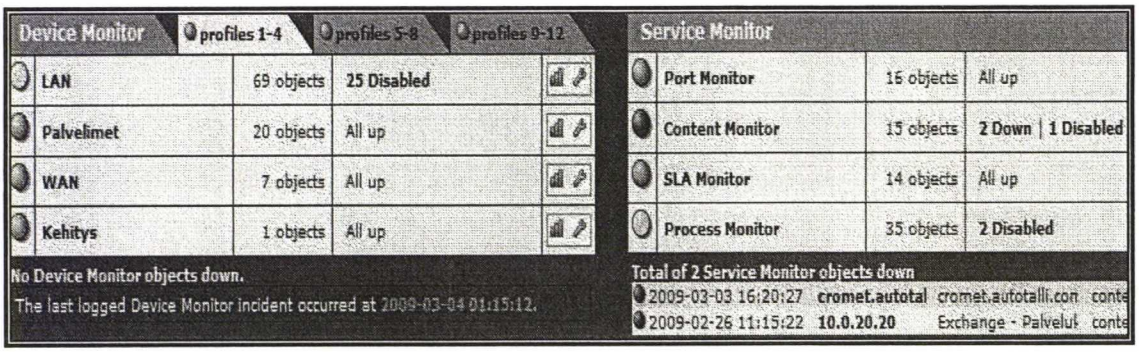
## 7.2 NetEye

NetEye on kaupallinen verkonvalvonta-ohjelmisto, jonka toteutus koostuu ohjelmisto- ja laitteistopohjaisesta ratkaisusta [14–15, 45]. Arkkitehtuuriratkaisultaan se on keskitetty verkonvalvonta-ohjelmisto. Toimitettava laiteratkaisu sisältää kaikki valvonnassa tarvittavat ohjelmat, poislukien agentti ohjelmiston. Agentti-ohjelmisto asennetaan valvottavilla laitteille, mikäli sen järjestelmälokitietoja halutaan välittää NetEye valvonta-ohjelmalle. Laitteiston käyttöönoton arviointi ei kuulunut diplomityön sisältöön. Koska NetEye käyttää verkonvalvontaan standardiprotokollia, kuten SNMPv2 ja ICMP, voidaan valvonta- ja raportointitietoja kerätä automaattisesti eri valmistajien laitteista ja palveluista sekä halutuista sovelluksista ja prosesseista. Ohjelmisto käyttää SNMP-protokollan haku-operaatiota valvontatietojen hakuun, muun muassa prosessorin kuormituksesta ja vapaana olevan keskusmuistin määrästä [45]. Lisäksi verkkolaitteista voidaan lukea verkkoliikenteeseen liittyviä parametrien arvoja. Luettujen tietojen perusteella muodostetaan kuvaajia sekä raportteja, joiden avulla mahdolliset poikkeamat verkkoliikenteessä tai siihen liittyvissä palveluissa voidaan havaita helpommin (kuva 22) [34, 42,45].




**Kuva 22: NetEye palveluseuranta laitekohtaisesti**

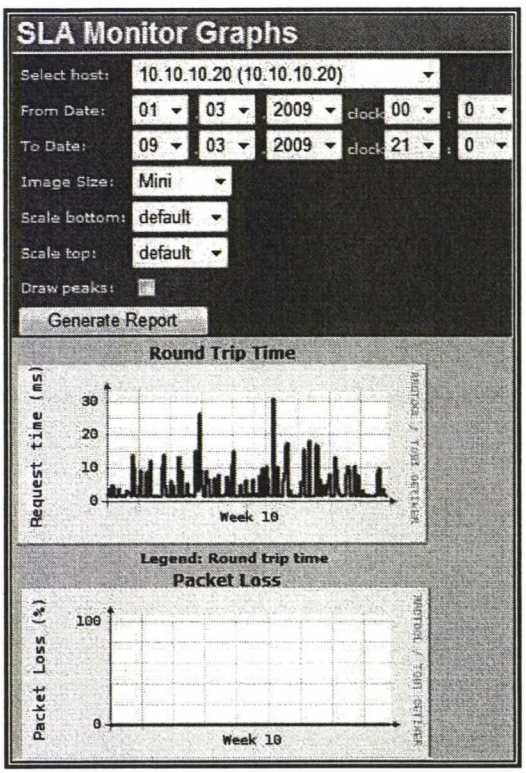
NetEye-ohjelmistoa, kuten omaa verkonvalvonta-työkalua ja HP SIM-ohjelmistoa, käytetään graafisen käyttöliittymän kautta. Graafisen käyttöliittymän aloitussivulla on yleisnäkymä valvottavien laitteiden, palveluiden, sovellusten ja prosessien toimivuudesta sekä tiedot valvonta-ohjelmistolle välitetyistä vika- ja virheviesteistä (kuva 4). Valvottavien palveluiden, laitteiden ja sovellusten tila kuvataan käyttöliittymässä liikennevalojen mukaisilla värisymboleilla sekä tekstimuodossa (kuva 4, 23).




**Kuva 23: Valvottavat laitteet ja palvelut kategorisoituna**

Laitteiden toimivuuden tarkistus suoritetaan periodisilla ICMP Ping-komennoilla, joiden vastausten perusteella valvottavan laitteen tila kuvataan käyttöliittymässä liikennevalojen mukaisilla värisymboleilla (kuva 23). Laitteiden ja valvottavien palveluiden kategorisointi erilaisiin ryhmiin on mahdollista. Näin ollen hallinta-ohjelman etusivulta nähdään kerralla kaikkien ryhmän laitteiden ja palveluiden tilatieto (kuva 4,23).

NetEye:ssä palveluiden valvonta on jaoteltu neljään kategoriaan. Porttitason valvonnassa lähetetään kolmivaiheinen TCP SYN kättely-viesti valvottavaan sovellusporttiin. Mikäli valvottava sovellus ei vastaa tähän viestiin SYN-ACK viestillä, raportoidaan sovelluksen toimimattomuus hallinta-sovelluksessa. Sisältövalvonta lähettää valvottavalle palvelulle etukäteen määritellyn kyselyviestin. Mikäli kyselyyn saatu vaste ei ole ennalta määritellyn vasteen mukainen, ei valvottava palvelu toimi oikein. Sisällönvalvonnan osalta NetEye:n on määritelty valmiiksi seuraavat sovellukset ja niiden oletusvasteita: WWW, SMTP, DNS, Telnet, FTP, LDAP. Oletusvasteita muokkaamalla sisällönvalvonta on nopeasti muokattavissa useisiin tarkoituksiin. Palveluntasovalvonta mittaa valvottavan tietoverkon toimivuutta lähettämällä valvottaville verkkolaitteille periodisesti ICMP echo-reply paketteja. Saatujen vastauksien perusteella lasketaan ja muodostetaan verkonvaste-aikaa kuvaava latenssi ja testeissä hävinneet paketti-kuvaajat (Kuva 24) [45]. SNMPv2 kyselyjä käytetään apuna, jotta prosessivalvonnan tiedot laitteiden prosessien ja palveluiden toiminnasta saadaan selville (Kuvat 23, 25).




**Kuva 24: SLA-raportti vasta-ajan osalta**

| Service monitor | | |
|---|---|---|
| **Port Monitor** | | |
| 10.10.10.20 (ldapssl) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| **Content Monitor** | | |
| SMTP (content-smtp) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| **SLA Monitor** | | |
| 10.10.10.20 (https) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| **Process Monitor** | | |
| Mailserver store.exe:store.exe (process-monitor: store.exe) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver Microsoft Exchange Information Store:Microsoft Exchange Information Store (windows-service) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver Microsoft Exchange Routing Engine:Microsoft Exchange Routing Engine (windows-service) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver Microsoft Exchange System Attendant:Microsoft Exchange System Attendant (windows-service) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver SMTP:Simple Mail Transfer Protocol (SMTP) (windows-service) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver World Wide Web Publishing Service:World Wide Web Publishing Service (windows-service) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver store.exe:store.exe (process-monitor: store.exe) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver DCHP:DHCP Server (windows-service) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |
| Mailserver DNS:DNS Server (windows-service) | up: 1 days 00:00:00 (100.0%) | down: 00:00:00 (0.0%) |

**Kuva 25: NetEye palveluseuranta laitekohtaisesti**

NetEye:n voidaan määritellä erillisiä hälytysryhmiä, jolle välitetään automaattisesti tieto, mikäli valvonnan piirissä olevissa palveluissa tai toiminnoissa havaitaan esimerkiksi virhetilanne. Tieto voidaan lähettää joko teksti-, sähköposti tai SNMP Trap viesteillä erilaisten ennalta määriteltyjen sääntöjen mukaisesti. Tietojen lähettämistä voidaan rajoittaa vastaanottajan, viikonpäivän ja kellonajan mukaan, esimerkiksi lähettämällä

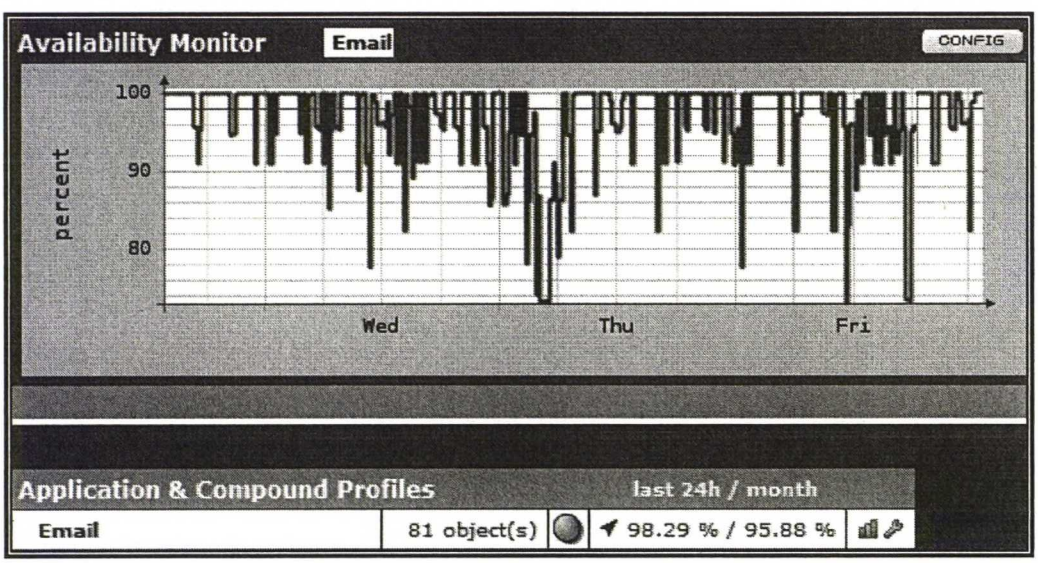
tieto tietystä viasta ainoastaan normaalin työajan ulkopuolella tietylle henkilölle tekstiviestinä.

Tietoturvaan liittyvät asiat on huomioitu kattavasti: Valvonta-ohjelmiston toteutus sisältää roolipohjaisen käyttäjäryhmittelyn, jonka avulla on mahdollista rajoittaa järjestelmän käyttöä kahteen käyttäjäryhmään, pää- ja normaalikäyttäjiin. Normaalikäyttäjien oikeudet ovat rajattu kattamaan vain verkonvalvontaan ja raportointiin liittyvät toiminnot. Tietoliikenne selainpohjaisen käyttöliittymän ja hallinta-ohjelmiston välillä voidaan salata SSL tekniikkaa käyttäen [45].

NetEye ohjelmistoon on lisätty verkonvalvontaan useasti tarvittavia aputyökaluja, kuten Ping ja Traceroute, joiden avulla erilaisten palveluiden tai laitteiden toimivuus voidaan tarkistaa nopeasti. Apuohjelmia voidaan käyttää muun muassa valvonta-ohjelmiston etusivun valikkojen kautta.

Raportoinnin osalta tuote sisältää kattavan valikoiman valmiita raporttimalleja. Näiden avulla palveluita valvova henkilö voi muodostaa esimerkiksi palvelukohtaisia SLA (Service Level Agreement) palvelutasoraportteja. SLA-raporttien avulla voidaan todentaa, ovatko valvottavalle palvelulle asetetut vaatimukset toteutuneet esimerkiksi saatavuuden, käytettävyyden tai kuormitettavuuden osalta (Kuva 24).

NetEye verkonvalvonta-ohjelmistoon voidaan integroida BAM- ja Analyze Engine-lisämodulit. BAM:n avulla voidaan seurata yhden tai useamman valvottavan järjestelmän kokonaiskäytettävyyttä reaaliajassa käytettävyysraporttien avulla (Kuva 26).

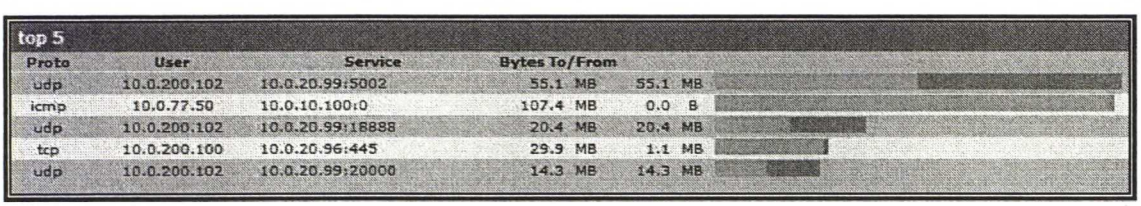



**Kuva 26: NetEye BAM SLA-valvonta**

Kaikista järjestelmään liittyvistä määritellyistä kokonaisuuksista kerätään valvontatietoja ja lopullinen käytettävyyskuvaaja muodostetaan annettujen komponenttien painoarvojen mukaisesti (Kuva 27).


**Kuva 27: NetEye BAM osa-järjestelmien painotus**

Analyze Engine-lisämodulia käytetään valvonnan avulla kerättyjen tietojen tarkempaan analysointiin. Sen avulla voidaan raportoida esimerkiksi tietoliikennekapasiteetin käyttöä protokolla- ja kohdetasoilla (kuva 28).




**Kuva 28: NetEye Analyze Engine**

## 7.3 Vertailu: Oma verkonvalvonta-ohjelmisto, HP SIM ja NetEye

Kaikki vertailtavat valvonta-ohjelmistot toimivat automaattisesti ja niiden käyttöön ei normaalitilanteessa tarvitse puuttua. Toteutukseltaan kaikki sisältävät sekä käyttäjätunnistuksen että graafisen selainpohjaisen etäkäyttöliittymän. Etäkäyttöliittymät on toteutettu niin, että valvontaa suorittava henkilö havaitsee helposti valvottavissa laitteissa, palveluissa tai sovelluksissa esiintyvät virhetilat käytettyjen värisymbolien avulla. Jokaisessa ohjelmistossa etäkäyttöliittymän kautta suoritetaan kaikki valvontaan tehtävät muutokset, kuten valvottavien laitteiden lisäys ja poisto. Vastaavasti etäkäyttöliittymän kautta suoritetaan raportointiin liittyvät toiminteet. Kaikissa vertailtavissa ohjelmistoissa oli mahdollisuus välittää tieto valvottujen laitteiden tai palveluiden viasta määritellyille vastaanottajille vähintään sähköpostiviestinä.

Itse kehitetyn verkonvalvonta-ohjelmiston suurimpina etuina ovat sen nopea ja yksinkertainen toiminta. Sen huonoina puolina ovat automaattisen toimintojen puuttuminen, joten kaikki etäkäyttöliittymän kautta tehtävät muutokset suoritettava käsin, kuten laitteen lisäys valvontaan. Lisäksi erilaisten automaattitoimintojen tai järjestelmätason pienet muutokset vaativat muutoksia suoraan ohjelmakoodiin. Vertailtaessa omaa ja HP SIM verkonvalvonta-ohjelmistoja, voidaan todeta että oman

ohjelmiston kautta voidaan valvoa laitteiston toimintaa vastaavalla tasolla kuin HP:n ohjelmistollakin.

Ilmainen HP SIM on valvonta- ja hallinta-ohjelmisto, jota voidaan käyttää myös verkonvalvontaan. Sen avulla voidaan nähdä nopeasti valvottavien laitteiden toimintatila, mutta sen avulla ei voida valvoa palveluita, sovelluksia ja prosesseja tai niiden toimintaa. Verrattuna NetEye ohjelmistoon HP SIM ei kerää esimerkiksi verkkolaitteiden verkkokorttitasolla liikennemittausta varten. Parhaiten SIM sopii sellaisen verkon valvontaan ja hallintaan, joka sisältää HP:n omia laitteita. Lisäksi ohjelmiston integrointimahdollisuus muihin verkonvalvonta-ohjelmistoihin, kuten HP Open View:n, tarjoaa kattavat työkalut.

NetEye on verkonvalvontaan tehty kaupallinen tuote, joka sisältää erittäin paljon toiminnallisuutta erilaisten laitteiden, palveluiden, sovellusten ja prosessien valvontaan. Lisäksi valvottavien palveluiden toiminnasta voidaan helposti tuottaa kattava raportointi ohjelmiston avulla muun muassa palveluntasoseurantaa varten.

Kuvassa 29 on listattu vertailtavien verkonvalvonta-ohjelmistojen suurimmat erot.

| **Vertailtava ominaisuus** | **Vertailtavan tuotteen nimi** | | |
|---|---|---|---|
| | **Oma verkonvalvonta -ohjelmisto** | **HP SIM** | **NetEye** |
| Asennus on paketoitu ohjelmistotuotteeksi | Ei | Kyllä | Kyllä |
| Vaaditaanko asennuksessa erillis- tai varusohjelmistoja | Kyllä | Ei | Ei |
| Tuki erilliselle tietokanta-ohjelmistolle | Ei | Kyllä | Ei |
| Käyttäjätunnistus | Kyllä | Kyllä | Kyllä |
| Etäkäyttöliittymän salattu tietoliikenne | Ei | Kyllä | Tarvittaessa |
| Pääsynvalvonnan eri tasot | Ei | Kyllä | Kyllä |
| SNMP-tuki versiolle | 1,2 | 1,2,3 | 1,2 |
| Laitteiden valvonta | Kyllä | Kyllä | Kyllä |
| Palveluiden valvonta | Kyllä | Ei | Kyllä |
| Sovellusten valvonta | Kyllä | Ei | Kyllä |
| Prosessien valvonta | Ei | Ei | Kyllä |
| Tuotteen hinta | Ilmainen | Ilmainen | Maksullinen |
| Raportointi: Sisältyy tuotteeseen | Osittain | Osittain | Kyllä |
| Raportointi: Laitetasolla kattavat raportit | Ei | Kyllä | Kyllä |
| Vikailmoitus kaikissa vikatilanteissa | Kyllä | Kyllä | Kyllä |
| Vikailmoitus määritellyissä vikatilanteissa | Osittain | Kyllä | Kyllä |
| Soveltuu proaktiiviseen valvontaan | Ei | Osittain | Kyllä |
| Soveltuu reaktiiviseen valvontaan | Kyllä | Kyllä | Kyllä |
| Laitetasolla kattavat raportit | Ei | Kyllä | Osittain |

**Kuva 29: Vertailu: Oma verkonvalvonta-ohjelmisto, HP SIM, NetEye**

## 7.4 Kehityskohteet

Diplomityön toteutuksen aikana listasin seuraavat verkonvalvonta-ohjelmiston kehityskohteet, jotka toteuttamalla ohjelmasta tulisi huomattavasti kehittyneempi:

Tekstitiedostojen korvaaminen relaatiotietokannalla, jolloin tallennettujen tietojen jatkojalostus ja -käyttö raportointiin tai analysointiin olisi tehokkaampaa ja helpompaa. Lisäsyynä relaatiotietokannan käyttöön ovat mm atomiset operaatiot, joilla varmistetaan tietokannan tilan eheys. Edellä kuvatuista syistä johtuen kaikissa vertailtavissa tuotteissa on relaatiotietokanta käytössä.

Nykyisten kiinteästi määriteltyjen valvottavien palveluiden, kuten FTP portissa 21, sijaan käyttäjälle annettaisiin mahdollisuus valita mitä palveluita tai portteja (socket) laitteen osalta halutaan valvoa. Tällä saataisiin tarjottua käyttäjälle mahdollisuus valvoa ohjelmiston avulla mitä tahansa palvelua ja sen toimivuutta.

CGI-optimointia tulisi tehdä, mikäli verkonvalvontatuotteella on paljon käyttäjiä ja valvottavia laitteita. CGI-tekniikalla toteutettu ohjelma käynnistetään joka kerran suoritettavaksi ja palvelimen kuormitus kasvaa nopeasti. Yksi optimointiratkaisu olisi käyttää Apache-palvelinohjelmistossa Mod_perl Perl-kirjastoa, "joka liittää yhteen Apache-palvelimen C-kielisen API-rajapinnan ja Perl-kielen. Tällöin Perlillä voi kirjoittaa Apache palvelimeen API-ohjelmia. Yleensä CGI-ohjelmiin verrattuna nopeudet nousevat paljon (jopa10-20 kertaisiksi)" [24]. Lisäksi Apache-palvelinohjelmiston prosessien ja säikeiden määrittelyä optimoimalla voitaisiin ohjelmiston suorituskykyä parantaa entisestään [46–47].

Raportteja voidaan monipuolistaa tallentamalla valvottavien laitteiden, sovellusten ja palveluiden tiedot useaan RRDtool-tietokantaan. Näistä tietokannoista voidaan hakea raportteja esimerkiksi histogrammia viimeisen vuoden ajalta erilaisilla kuvaajilla.

Tietoturvan liittyviä parannuksia olisi SNMP version kolme käyttöönotto sekä etäkäyttöliittymässä käytettävän HTTP-yhteyden korvaaminen HTTPS:llä. Jälkimmäisen toteuttaminen vaatisi sertifikaatin asennuksen Apache WWW-palvelimeen sekä pieniä muutoksia Apache-ohjelmiston asetuksiin [46–47].

# 8 Tulokset ja yhteenveto

Diplomityössäni proof-of-concept verkonvalvonta-ohjelmiston toteutus onnistui sillä kaikki vaatimusmäärittelyn sisältämät toiminnat sekä toiminnallisuudet sisältyivät valmiiseen ohjelmistoon.

Vertailu kaupallisiin ja avoimeen lähdekoodiin perustuviin tuotteisiin paljasti kuitenkin sen, miten paljon lisätoiminnallisuutta vertailtavissa tuotteissa on esimerkiksi raportoinnin osalta valmiina. Toisaalta vertailussa kävi ilmi, että yksinkertaisilla ja nopeasti toteutettavissa olevilla verkonvalvonta-ohjelmistoilla pystytään toteuttamaan rajoitetusti useat kaupallisissa tuotteissa olevat toiminnallisuudet. Lisäksi molemmista vertailtavista verkonvalvonta-ohjelmista löytyi sellaisia tarvittavia ominaisuuksia ja toimintoja, joita toisessa ei ollut.

Kaikkien kaupallisissa tai avoimeen lähdekoodiin perustuvissa tuotteissa olevien toimintojen toteuttaminen omalla verkonvalvontatuotteella ei ole järkevää. Mikäli lisätoimintoja tarvitaan, niin vaihtoehtoina olisi joko jatkaa omaa verkonvalvonta-ohjelmiston kehittämistä yhdistämällä siihen jo olemassa olevia avoimen lähdekoodin lisäosia esimerkiksi raportoinnin, tiedonkeruun ja vikaviestien välittämisen parantamiseksi

Verkonvalvontatyökalun toteuttaminen oli haastava ja mielenkiintoinen tehtävä, ja sen toteuttamien antoi hyvän yleiskuvan mitä ohjelmistonkehitys sisältää kokonaisuutena. Työn edistyessä yleiskuva verkonvalvonta-ohjelmiston toiminnasta, toteutuksesta ja siihen liittyvistä haasteista selkeytyi. Laadunvalvonnan, testaamisen ja dokumentaation merkitys ohjelmistoprojektin läpiviemisen sekä jatkokehityksen kannalta konkretisoituivat sekä arvioit niihin oikeasti tarvittavasta ajasta ja resursseista.

# 9 Liitteet

## Liite 1: RRDtool-esimerkki

```
sub rrdpic($filename,$test)
{
	my $systemtime_file;
	my $rrd_db_filename = "$filename".".rrd";
	my $rrd_pic_filename = "$filename".".png";
	my $rrd_cmd0="rrdtool";
	my $rrd_cmd1="create";
	my $rrd_cmd2="start";
	my $rrd_cmd3="end";
	my $rrd_cmd4="update";
	my $rrd_cmd5="fetch";
	my $rrd_cmd6="graph";
	my $starttime;
	my $endtime;
	my @store;
	my @store2;
	my $apu;
	my $apu2;
	my $testtt=$test;
	$systemtime_file="result-systemtime-dippa.txt";
	$apu=0;
	@args=("del $rrd_db_filename");
	system (@args)==0 or die "Cannot delete RRD-database $rrd_db_filename:
	system @args failed\n";
	open CONF, '<', $systemtime_file;
	while (<CONF>)
	{
		chomp;
		@store[$apu]=$_;
		$starttime=@store[0];
		$endtime=@store[$apu];
		$apu++;
	}
	@args=("$rrd_cmd0 $rrd_cmd1 $rrd_db_filename --$rrd_cmd2 $starttime
	DS:response:GAUGE:300:0:10000 RRA:LAST:0.5:1:24
	RRA:AVERAGE:0.5:6:10");
	system (@args)==0 or die "Cannot create RRD-database $rrd_db_filename:
	system @args failed\n";
	$filename="$filename".".txt";
	open INFO, '<', $filename or die "Cannot open file $filename\n";
```

```
readline(INFO);
while (<INFO>)
{
        chomp;
        @store2[$apu2]=$_;
        @args=("$rrd_cmd0 $rrd_cmd4  $rrd_db_filename @store2[$apu2]");
        system (@args)==0 or die "Cannot update RRD-database
        $rrd_db_filename: system @args failed\n";
        $apu2++;
}
if (-e $rrd_pic_filename)
{
        @args=("del $rrd_pic_filename");
        system (@args)==0 or die "Cannot delete RRD-picture $rrd_pic_filename:
                system @args failed\n";
}
my $title='': ';
my $title2=' Response graph';
my $title3='';
@args = ("$rrd_cmd0 $rrd_cmd6 $rrd_pic_filename --$rrd_cmd2 $starttime --end
$endtime --title $title $test $hostipaddr $title2 $title3 --vertical-label ms,
DEF:myresponse=$rrd_db_filename:response:LAST
LINE2:myresponse#FF0000");
system (@args)==0 or die "Cannot create RRD-picture  $rrd_db_filename:
system          @args failed\n";
}
```

## Liite 2: Verkonvalvonta-ohjelmiston hakemistot ja ohjelmat

| Hakemisto tai tiedosto | Selitys |
|---|---|
| C:\Program Files (x86)\Apache Software Foundation\Apache2.2\cgi-bin | Verkonvalvonta-ohjelmiston CGI-tiedostot (*.pl ja *.cgi) |
| Main.pl | Aloitus- ja pääsivu: Hakee tekstitiedostoista valvottavat laitteet, palvelut ja sovellukset. Näyttä graafisesti näiden tilatiedon yleistasolla sekä valvontaspesifisesti: Vihreä (toimii), punainen (ei toimi) tai harmaa (valvontaa ei ole määritelty päälle).<br>Sivulta voidaan siirtyä editointi - ja raportointisivulle |
| Report.cgi | Raportointisivusto. Lukee valvottavien laitteiden, palveluiden ja sovellusten raportointitiedot tekstitiedostoista ja näyttää nämä kuvina sekä SNMP-tiedot tekstinä. Sivulta voidaan siirtyä pää- ja editointisivuille. |
| Edit.pl | Editointisivu: Editointisivulta voidaan lisätä, poistaa ja muokata valvottavia laitteita, palveluita ja sovelluksia. Editointisivulta voidaan siirtyä pääsivulle |
| Save.cgi | Muokkaa config.txt tekstitiedostoa editointisivulla tehtyjen muutosten, poistojen tai lisäysten mukaisesti sekä päivittää tiedot sivun listaan tallennuksen jälkeen |
| Post.cgi | Kun uusi valvottava laite lisätään Editointi-sivulla, post.cgi lukee laitteen tiedot ja tallentaa ne dippa_config.txt tekstitiedoston sekä päivittää muutokset Editointi-sivun laitelistaan |
| C:\Program Files (x86)\Apache Software Foundation\Apache2.2\conf | Apache ohjelmiston määrittelyt tiedostot |
| httpd.conf | Käyttäjätunnistukseen tarvittavat määritykset |
| C:\Program Files (x86)\Apache Software Foundation\Apache2.2\htdocs | Verkonvalvonta-ohjelmiston käyttämät kuvatiedostot: Vihreä (toimii), punainen (ei toimi) tai harmaa (valvontaa ei ole määritelty päälle). |
| C:\Program Files (x86)\Apache Software Foundation\Apache2.2\htdocs\images | Ajastetut tehtäväskriptit, raportointitiedostot, tietokannat ja verkonvalvonta-ohjelmisto |
| network.pl | Verkonvalvonta-ohjelmisto, joka sisältää laitteiden, palveluiden ja sovellusten valvontatoimitteet |
| systemtime.pl | Verkonvalvonta-ohjelmiston apu-ohjelma, joka hakee ja tallentaa järjestelmäajan EPOC muodossa |
| config.txt | Verkonvalvonta-ohjelmiston valvontalistaus: Sisältää valvottavien laitteiden, palveluiden ja sovellusten tiedot |
| result-systemtime.txt | Systeemiajan tallennustiedosto, jota systemtime.pl |

| | käyttää |
|---|---|
| network.bat | Ajastettu skripti, joka käynnistää network.pl verkonvalvonta-ohjelmiston |
| systemtime.bat | Ajastettu skripti, jonka käynnistää systemtime.pl systeemiajan haku-skriptin |

# 10 Viittaukset


1. Comer, Internetworking with TCP/IP: Principles, Protocols and Architectures, Prentice Hall, ISBN: 0-13-018380-6

2. L. Stein, CGI, CPAN, 2009, http://search.cpan.org/~lds/CGI.pm-3.42/CGI.pm

3. T. Oetiker, About RRDtool, 2008, http://oss.oetiker.ch/rrdtool/index.en.html

4. T. Oetiker, RRDtool Documentation, 2008, http://oss.oetiker.ch/rrdtool/doc/index.en.html

A. Bogaerdt, RRDtutorial, 2008, http://oss.oetiker.ch/rrdtool/tut/rrdtutorial.en.html

5. K. Patel, RRD-beginners, 2008, http://oss.oetiker.ch/rrdtool/tut/rrd-beginners.en.html

6. T.Oetiker, RRDtool binary, rrdtool-1.2.28-bin-w32.zip, 2008, http://www.gknw.net/mirror/rrdtool/

7. Hewlett-Packard, HP Systems Insight Manager 5.2 Installation and Configuration Guide for Microsoft Windows, Hewlett Packard, 2008, http://docs.hp.com/en/418812-004/418812-004.pdf

8. Hewlett-Packard, HP Systems Insight Manager 5.2 Release Notes, Hewlett Packard, 2008, http://www.docs.hp.com/en/431828-004/431828-004.pdf

9. Hewlett-Packard, HP Systems Insight Manager: Quick Specs, 2008, http://h18013.www1.hp.com/products/quickspecs/11824_div/11824_div.PDF

10. Hewlett-Packard, Insight Management MIB update kit for HP Systems Insight Manager for Windows, 2008, http://h20000.www2.hp.com/bizsupport/TechSupport/SoftwareDescription.jsp?lang=en&cc=US&swItem=MTX-36ac981a5a4a4a349b41f4669e

11. Hewlett-Packard, SIM 5.2 with SP2, 2008, http://h20293.www2.hp.com/portal/swdepot/try.do?productNumber=HPSIM-Win

12. Hewlett-Packard, HP Systems Insight Manager 5.2 Technical Reference Guide, 2008, http://docs.hp.com/en/356920-401/356920-401.pdf

13. Noval NetEye, Noval Networks, 2008, http://www.novalnetworks.com/neteye

14. Noval NetEye demo, Noval Networks, 2009, http://www.neteye.fi/

15. Open NMS Documentation, The OpenNMS Group, 2009, http://www.opennms.org/index.php/Docu-overview

16. S.Tripathi, Windows, Apache and .htaccess authentication, Sniptools, 2009, http://sniptools.com/vault/windows-apache-and-htaccess-authentication

17. Anonymous, Maximum Linux Security, SAMS, 1999, ISBN: 0-672-316708-6

18. Netcraft, December 2008 Web Server Survey, 2008, http://news.netcraft.com/archives/web_server_survey.html

19. D. Blank-Edelman, Perl for System Administration, O'Reilly, 2000, ISBN: 1-56592-609-9

20. J. Friedl, Mastering Regular Expressions, O'Reilly, 1997, ISBN: 1-56592-257-3

21. L. Wall, T. Christiansen, R. Schwartz, Programming Perl, O'Reilly, 1996, ISBN: 1-56592-149-6

22. E.Castro, Perl ja CGI, IT Press, 2000, ISBN: 951-826-219-5

23. J.Peltomäki, V. Inkinen, A.Rantala, CGI- ja ASP-ohjelmointi, Teknolit, 2000, ISBN: 951-846-037-X

24. X. Faulkner, Usability Engineering, Palgrave, 2000, ISBN: 0-333-77321-7

25. I.Haikala, J.Märijärvi, Ohjelmistotuotanto, Suomen ATK-kustannus, 1998, ISBN: 951-762-666-5

26. A. Ganot, SendMail, 2008, http://www.petri.co.il/software/sendmail.zip

27. D. Town, Net::SNMP, CPAN, 2009, http://search.cpan.org/~dtown/Net-SNMP-5.2.0/lib/Net/SNMP.pm

28. W. Stallings, SNMP, SNMPv2, SNMPv3, and RMON 1 and 2. Addison Wesley Longman, Inc, 3rd edition 1999.

29. L. Li, M. Thottan, B. Yao, S. Paul, 2003, Distributed Network Monitoring with Bounded Link Utilization in IP networks, IEEE Infocom 2003, http://www.cs.cornell.edu/lili/liliINFOCOM03.pdf

30. H. Ballani, P. Francis, CONMan: Taking the Complexity out of Network Management, 2006, SIGCOMM 06 Workshop September 15.11.2006, http://www.cs.cornell.edu/~hitesh/pubs/inm06-conman.pdf

31. K. Fukuda, K. Cho, H. Esaki, The Impact of Residential Broadband Traffic on Japanese ISP Backbones, 2004, http://www.iijlab.net/~kjc/papers/ivs-rbb-traffic.pdf

32. R. Craig, S. Jaskiel, Systematic Software Testing, Artech House, 2002, ISBN 978-1-58053-508-3

33. B. Siaterlis, B. Maglaris, Detecting DDoS attacks with passive measurement based heuristics, Computers and Communications, 2004. Proceedings. ISCC 2004, Ninth International Symposium on Volume 1, Issue, 2004.

34. D. Harrington, R. Presuhn, B. Wijnen, An Architecture for Describing, SNMP Management Frameworks, Network Working Group
Request for Comments: 2271, 01.1998, http://www.isi.edu/in-notes/rfc2271.txt

35. U. Blumenthal, B. Wijnen, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3), Request for Comments: 3414, December 2002, http://www.rfc-archive.org/getrfc.php?rfc=3414

36. F. Maino, The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model, 06.2004, IETF,
http://www.ietf.org/rfc/rfc3826.txt

37. S.Aly, H.Abu-Salem, M.Abu-Kreisha, On the Multicasting Security of SNMPv3, http://www.iseing.org/emcis/EMCIS2005/pdfs/33.pdf]

38. R. Brown, C. McMillen, NET::Ping, CPAN, 2009,
http://search.cpan.org/~smpeters/Net-Ping-2.35/lib/Net/Ping.pm

39. P. Yalagandula, P. Sharma, S. Banerjee, S. Basu, S. Lee, A Scalable Sensing Service for Monitoring Large Networked Systems, Sigcomm, SIGCOMM 06 Workshop September 15.11.2006,
http://www.comsoc.org/confs/infocom/2006/Posters/1568980849_S%20%20ScalableNetworkSensingService.pdf

40. A.Ali, F.Nazir, H.Burki, M Tarar, I. Legrand, Constella: IP Network Topology Discovery for Large and Multi-Subnet Networks, 2004, http://www.comtec.e-technik.uni-kassel.de/ICOMP/IC2004/ConfMan/SUBMISSIONS/39-awtitefzir.pdf

41. Z. Wang, Q. Xia, K. Lu, Two-Tier GCT Based Approach for Attack Detection, J. Software Engineering & Applications, 2008,
http://www.scirp.org/Journal/PaperDownload.aspx?paperID=148&fileName=JSEA20080100010_65705794

42. L. Wenwei, Z. Dafang, Y. Jinmin, X Gaogang, Computer communications, 2007, http://www.ict.ac.cn/grope/down/07-12/1197533607.pdf

43. G. Barr.version 1.31, CPAN, 2009, http://search.cpan.org/~gbarr/IO-1.2301/IO/Socket/INET.pm

44. S. Harris, CISSP All in One Exam Guide 4th edition, MC Graw Hill, 2008, ISBN 978-0-07-149787-9

45. Noval Networks, Noval NetEye Base 3.4.9, Administrator Manual, 2009, http://www.neteye.fi/share/docs/Base_v3.4.9_adm_guide_public.pdf

46. K. Coar, R. Bowen, Apache Cookbook, O'Reilly, 2007, ISBN 978-0-59-652994-9

47. B.Laurie, P. Laurie, Apache The Definite Guide, 3rd Edition, O'Reilly, 2002, ISBN 0-596-00203-3

# 11 Lyhenteet

| | |
|---|---|
| 3-DES | Triple Data Encryption Standard |
| AES | Advanced Encryption Standard |
| ASN.1 | Abstract Syntax Notation One |
| BIOS | Basic Input Output System |
| BER | Basic Encoding Rules |
| CGI | Common Gateway Interface |
| CM | Central Manager |
| DES | Data Encryption Standard |
| DNS | Domain Name Service |
| HP | Hewlett Packard |
| ICMP | Internet Control Message Protocol |
| IP | Internet Protocol |
| FTP | File Transfer Protocol |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | Hypertext Transfer Protocol Secure |
| MAC | Media Access Control |
| MD5 | Message Digest |
| MIB | Management Information Base |
| NetBIOS | Network Basic Input Output System |
| NMS | Network Management System |
| OID | Object Identifier |
| OSI | Open Systems Interconnection |
| PDU | Protocol Data Unit |

| | |
|---|---|
| PERL | Practical Extraction and Report Language |
| PPM | Perl Package Manager |
| RMON | Remote Network Monitoring |
| RRDTool | Round Robin Database Tool |
| SIM | System Insight Manager |
| SHA1 | Secure Hash Algorithm |
| SLA | Service Level Agreement |
| SMI | Structure of Management Information |
| SMTP | Simple Mail Transfer Protocol |
| SNMP | Simple Network Monitoring Protocol |
| SSL | Secure Sockets Layer |
| SQL | Structured Query Language |
| TCP | Transmission Control Protocol |
| TLS | Transport Layer Security |
| UDP | User Datagram Protocol |
| WWW | World Wide Web |